10-05-00

A

NEW APPLICATION TRANSMITTAL

ASSISTANT COMMISSIONER FOR PATENTS
Box Patent Application
Washington, D.C. 20231

PATENT
Attorney Docket No.: 19880-002810

Dear Sir or Madam:

Transmitted herewith for filing under 37 CFR §1.53(b) is a non-provisional patent application of:

Applicant(s): DONALD F. GORDON et al.
For: METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS

Enclosed are:

Patent Application - 12 pages of Specification, 4 pages of Claims, 1 page of Abstract, 4 sheets of formal drawings, 104 pages for Exhibit A, and 52 pages for Exhibit B
An unsigned Declaration
An unsigned Statement to establish Small Entity Status under 37 CFR §1.9 and §1.27

APPLICANT REQUESTS DEFERRAL OF THE FILING FEE UNTIL THE RESPONSE TO THE NOTICE OF MISSING PARTS

PLEASE DO NOT CHARGE THE DEPOSIT ACCOUNT FOR THE FILING FEE

Please charge Deposit Account No 50-1229 as follows:
Any additional fees associated with this paper or during the pendency of this application.

Respectfully submitted,

Truong T. Dinh
Reg. No. 40,993

DINH & ASSOCIATES
2506 Ash Street
Palo Alto, California 94306
Tel: (650) 289-0600
Fax: (650) 289-0700

Express Mailing Label No.: EJ135053182US

Date of Deposit: 10/4/00

I hereby certify that this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR § 1.10 on the date indicated above and addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231

By: ______________________
Truong T. Dinh

VERIFIED STATEMENT (DECLARATION) CLAIMING SMALL ENTITY STATUS (37 CFR 1.9(f) & 1.27(c)) - SMALL BUSINESS CONCERN

Applicant or Patentee: **DONALD F. GORDON et al.**
Title: **METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS**
Application Serial No.: **Unassigned**
Filing Date: **10/4/00**

I hereby declare that I am:
[ ] the owner of the small business concern identified below:
[ ] an official of the small business concern empowered to act on behalf of the concern identified below:

Name of Small Business Concern: DIVA Systems Corporation
Address of Small Business Concern: 800 Saginaw Drive, Redwood City, CA, 94063

I hereby declare that the above-identified small business concern qualifies as a small business concern as defined in 13 CFR §121.12, and reproduced in 37 CFR §1.9(d), for purposes of paying reduced fees to the United States Patent and Trademark Office, in that the number of employees of the concern, including those of its affiliates, does not exceed 500 persons. For purposes of this statement, (1) the number of employees of the business concern is the average over the previous fiscal year of the concern of the persons employed on a full-time, part-time or temporary basis during each of the pay periods of the fiscal year, and (2) concerns are affiliates of each other when either, directly or indirectly, one concern controls or has the power to control the other, or a third party or parties controls or has the power to control both.

I hereby declare that rights under contract or law have been conveyed to and remain with the small business concern identified above with regard to the invention, entitled METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS by inventor(s) DONALD F. GORDON et al. described in Application Serial No. Unassigned, filed 10/4/00.

If the rights held by the above identified small business concern are not exclusive, each individual, concern or organization having rights in the invention is listed below* and no rights to the invention are held by any person, other than the inventor, who would not qualify as an independent inventor under 37 CFR §1.9(c) if that person made the invention, or by any concern that would not qualify as a small business concern under 37 CFR §1.9(d), or a nonprofit organization under 37 CFR §1.9(e).

*NOTE: Separate verified statements are required from each named person, concern or organization having rights to the invention averring to their status as small entities. (37 CFR §1.27)

Name: DIVA Systems Corporation
Address: 800 Saginaw Drive, Redwood City, CA, 94063
[ ] Individual [ ] Small Business Concern [ ] Nonprofit Organization

I acknowledge the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate. (37 CFR §1.28(b)).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application, any patent issuing thereon, or any patent to which this verified statement is directed.

Name of Person Signing: Stephanie Storms
Title of Person if other than Owner: Senior Vice President
Address of Person Signing: 800 Saginaw Drive, Redwood City, CA, 94063

Signature: ______________________ Date: ______________

Attorney Docket No.: 19880-002810
Client Reference No.: 275

PATENT APPLICATION

METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS

Inventors:

Donald F. Gordon, a citizen of United States, residing at,
170 Formway Court
Los Alto, CA 94022

Brian Feinberg, a citizen of United States residing at,
19782 Bixby Drive
Cupertino, CA 95014

Eugene Gershtein, a citizen of United States, residing at,
1418 Valota Road
Redwood City, CA 94061

Sadik Bayrakeri, a citizen of Turkey, residing at,
733 Shell Boulevard #104
Foster City, CA 94404

John P. Camito, a citizen of United States, residing at,
907 Pleasant Hill Road
Redwood City, CA 94061

Edward A. Ludvig, a citizen of Canada, Residing at,
931 Canyon Road
Redwood City, CA 94061

Assignee:

DIVA Systems Corporation
800 Saginaw Drive
Redwood City, CA 94063

Entity: Small business concern

DINH & ASSOCIATES
2506 Ash Street
Palo Alto, CA 94306
Tel (650) 289-0600
Fax (650) 289-0700

Attorney Docket No.: 19880-002810
Client Reference No.: 275

METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS

CROSS-REFERENCES TO RELATED APPLICATIONS

This application is a continuation-in-part of U.S. Patent Application Serial No. 09/466,990, entitled "STREAM INDEXING FOR DELIVERY OF INTERACTIVE PROGRAM GUIDE," filed December 10, 1999, which is a continuation-in-part of Serial No. 09/293,535, entitled "IMPROVED DATA STRUCTURE AND METHODS FOR PROVIDING AN INTERACTIVE PROGRAM GUIDE", filed April 15, 1999, Serial No. 09/384,394, entitled "METHOD AND APPARATUS FOR COMPRESSING VIDEO SEQUENCES," filed August 27, 1999, and Serial No. 09/428,066 filed October 27, 1999, entitled "METHOD AND APPARATUS FOR TRANSMITTING VIDEO AND GRAPHICS IN A COMPRESSED FORM."

This application is further a continuation-in-part of U.S. Patent Application Serial No. 09/539,228, entitled "MESSAGING PROTOCOL FOR DEMAND-CAST SYSTEM AND BANDWIDTH MANAGEMENT," filed March 30, 2000, which is a continuation-in-part of U.S. Patent Application Serial No. 09/524,854, entitled "BANDWIDTH MANAGEMENT TECHNIQUES FOR DELIVERY OF INTERACTIVE PROGRAM GUIDE," filed March 14, 2000.

The above-identified related applications are all assigned to the assignee of the present invention and incorporated herein by reference in their entirety for all purposes. Application Serial No. 09/466,990 is attached hereby as Exhibit A and Serial No. 09/539,228 is attached hereby as Exhibit B.

BACKGROUND OF THE INVENTION

The present invention relates to communications systems in general. More specifically, the invention relates to techniques to efficiently deliver interactive program guide (IPG) in a server-centric system.

Over the past few years, the television industry has seen a transformation in a variety of techniques by which its programming is distributed to consumers. Cable television systems are doubling or even tripling system bandwidth with the migration to hybrid fiber coax (HFC) cable plant. Customers unwilling to subscribe to local cable systems have switched in high numbers to direct broadcast satellite (DBS) systems. And, a variety of other approaches have been attempted focusing primarily on high bandwidth

digital technologies, intelligent two way set top terminals, or other methods of trying to offer service differentiated from standard cable and over the air broadcast systems.

With this increase in bandwidth, the number of programming choices has also increased. Leveraging off the availability of more intelligent set top terminals, several companies such as Starsight Telecast Inc. and TV Guide, Inc. have developed elaborate systems for providing an interactive listing of a vast array of channel offerings, expanded textual information about individual programs, and the ability to look forward to plan television viewing as much as several weeks in advance.

With this increase in the quantity of programming, it is a challenge to deliver program guide data to viewers in an efficient and effective manner. A large amount of resources (e.g., bandwidth) would normally be needed to continually transmit, for example, two weeks of programming for 200 channels. Therefore, efficient and effective techniques to deliver interactive program guide to a large number of viewers are highly desirable.

SUMMARY OF THE INVENTION

The invention provides various techniques to effectively and efficiently deliver interactive program guide. In accordance with the "multicasting" techniques of the invention, a varying number of transport streams can be generated and used to serve a distribution node having time varying demands. Multiple transport streams can provide additional transmission capacity (i.e., more bandwidth) and can also accommodate a larger number of packet identifiers (PIDs), which is especially useful for a demand-cast system during periods of heavy demands. The particular number of transport streams to be provided to the distribution node can be based on the actual needs of the node and, in accordance with an aspect of the invention, can be dynamically adjusted. Thus, additional transport streams can be sent to the distribution node as demands increase, with more transport streams being provided during periods of heavy demands. Correspondingly, transport streams can be tear down when the demands subside.

An embodiment of the invention provides a system for delivering interactive program guide (IPG). The system includes a number of encoding units, at least one transport stream generator, and a session manager. The encoding units encode a number of IPG pages and generate a number of (e.g., guide, video, audio, and data) streams, with each stream being assigned a respective packet identifier (PID). Each transport stream generator receives and multiplexes selected ones of the streams from one

or more encoding units into one or more transport streams. The session manager directs each transport stream generator to generate one or more transport streams based on usage. The system may further include a bandwidth manager that monitors usage and reports the usage to the session manager.

The encoding units can be operated to encode only once each IPG page to be transmitted from the system. Also, each encoding unit can perform slice-based or picture-based encoding of the IPG. Each transport stream generator can be operated to provide differentiated IPG for the neighborhood being served by the transport stream generator.

The number of transport streams generated by each transport stream generator can be dynamically adjusted based on demands from the neighborhood being served by the transport stream generator. Each transport stream generator can be directed to generate an additional transport stream if usage exceeds the capacity of the currently transmitted transport streams. For example, additional transport stream can be generated if the number of (e.g., guide, video, audio, and data) streams to be transmitted or if the required number of PIDs exceeds the capacity provided by the currently transmitted transport streams. Correspondingly, a particular transport stream can be tear down if usage falls below the capacity of remaining transport streams.

Various multiplexing structures can also be used, as described below. When multiple transport streams are employed, the IPG pages for the transport streams can be organized to reduce the likelihood of switching between transport streams at a receiving device, and to increase the likelihood of PID transitions within the same transport stream.

The invention further provides other systems and methods that implement, process, and/or facilitate the IPG delivery techniques described herein. Also, the techniques described herein can be used to deliver other types of contents besides IPG.

The foregoing, together with other aspects of this invention, will become more apparent when referring to the following specification, claims, and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings.

FIGS. 1A and 1B are block diagrams of two embodiments of an information distribution system that can be used to provide interactive program guide (IPG) and implement the multicasting of the invention;

FIG. 2A is a diagram of a first multiplexing structure wherein the IPG pages are provided within a single transport stream;

FIG. 2B is a diagram of a second multiplexing structure wherein the IPG pages are provided within multiple transport streams; and

FIG. 2C is a diagram of a third multiplexing structure wherein the IPG pages are provided within multiple transport streams with overlapping guide PIDs.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common within a figure.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

As shown in FIG. 1 of the attached Exhibit B, the head-end may service a number of distribution nodes directly or via local neighborhood equipment. Each distribution node may include a number of terminals (e.g., 4000 or more terminals). The programming (e.g., the IPG) provided to each distribution node may be different from that of other nodes. This differentiated programming can be achieved by transmitting one or more (distinct) transport streams to each distribution node.

As shown in FIG. 8 of Exhibit B, a number of IPG pages can be continually broadcast to each distribution node (e.g., 40 pages in the example shown in FIG. 8 of Exhibit B). Other IPG pages can be sent to viewers within the distribution node as requested via demand-cast. The IPG pages can be coded using picture-based and/or slice-based encoding schemes in the manner described in the attached Exhibit A and the coded pages can be assigned PIDs. Depending on the particular encoding scheme used, a set of 10 IPG pages may utilized 12 PIDs for the video and guide portions, another PID for the audio, and one or more PIDs for the data (as shown in FIG. 10C of Exhibit A). For demand-cast, each requested IPG page may be assigned with one or more PIDs for the demand-casted page.

To service a large number of terminals in a particular distribution node, especially during periods of heavy activity (e.g., during prime time periods) a large number of PIDs may be required. In accordance with the MPEG-2 standard, only a particular number of PIDs can be supported by each transport stream, as shown in FIG. 29 of Exhibit A. Also, depending on the demands, a large transmission capacity may be

required to send the required contents (e.g., the requested IPG pages). In a system in which resources (i.e., bandwidth) is limited, viewer requests for IPG pages may not be serviced if the required bandwidth is not available, which then results in blockage as described in Exhibit B. Blockage degrades the quality of service and is highly undesirable in many circumstances.

An aspect of the invention provides "multicasting" techniques that can be used to serve the time varying demands of a distribution node. Via multicasting, a number of transport streams can be generated and used to service a distribution node having, or during periods of, heavy demands. The multiple transport streams can provide additional transmission capacity (i.e., more bandwidth) and can also accommodate a larger number of PIDs. The larger number of PIDs is especially useful for demand-cast, during periods of heavy demands. The particular number of transport streams to be provided to the distribution node can be based on the actual needs of the node and, in accordance with an aspect of the invention, can be dynamically adjusted. Thus, additional transport streams can be sent to the distribution node as demands increase, with more transport streams being provided during periods of heavy demands. Correspondingly, transport streams can be tear down when the demands subside.

FIG. 1A is a block diagram of an embodiment of an information distribution system 100a that can be used to provide interactive program guide (IPG) and implement the multicasting of the invention. Distribution system 100a includes a head-end 102a, a number of distribution nodes 106, and a number of set top terminals 108 coupled to each distribution node.

In the embodiment shown in FIG. 1A, head-end 102a includes a session manager 112, a bandwidth manager 114, a bank of encoding and packetizing units 120, and a number of transport stream generators 130 (e.g., one transport stream generator for each distribution node being served). Head-end 102a receives contents from content sources 150, which may include a Video-on-demand (VOD) source 152a, an interactive program guide source 152b, a programming v4652c, an audio source 152d, a data source 152e, and other sources 152d for other types of content (e.g., advertisements, and so on). The contents are provided to the bank of encoding and packetizing units 120. Each encoding and packetizing unit 122 receives the designated contents (e.g., the guide and video portions for one or more IPG pages to be transmitted) and generates a number of streams, with each stream being assigned a respective PID.

For example, to encode ten IPG pages such as that shown in FIG. 10C in Exhibit A, encoding and packetizing unit 122 may receive ten video inputs for the ten IPG pages, one audio input, and one or more data inputs. Encoding and packetizing unit 122 encodes and packetizes the guide portion of each video input, the video portion of one of the video inputs, the audio input, and the data input(s). Encoding and packetizing unit 122 can then output ten guide streams, one video stream, one audio stream, and one or more data streams. Each guide, video, audio, and data stream is assigned a respective PID. Each IPG page can be encoded using a slice-based or picture-based encoding scheme, depending on the particular implementation of the encoding and packetizing unit.

Each transport stream generator 130 receives the outputs from one or more encoding and packetizing units 122 and multiplexes the received streams to form one or more transport streams to be provided to the distribution node. The multiplexing of the guide, video, audio, and data streams can be performed as described in Exhibit A. To form each transport stream, one packet of each of a number of video, audio, and data streams may be sequentially multiplexed into the transport stream. For example, a packet from each of guide streams 1 through 10 (e.g., for the ten IPG pages in FIG. 10C in Exhibit A), then a packet from a video stream, then a packet from an audio stream, then a packet from a data stream, and so on, can be multiplexed into the transport stream.

Transport stream generator 130 also provides packets conveying a program mapping table (PMT) for each final transport stream. The PMT specifies the PID values for program components. For example, a program may correspond to a particular broadcast channel, and the PMT may specify the PID values for the video, audio, and data streams relating to that broadcast channel.As noted above, each neighborhood served by a distribution node may have different program listings (e.g., different guide portions) but typically shares the same video(s). In addition, each neighborhood typically has different demand-cast requirements, which are dictated by the demands of the particular viewers in the neighborhood. During normal operation, the output stream from each transport stream generator 130, which includes one or more transport streams, will likely be different from the output streams from other transport stream generators for other distribution nodes.

Session manager 112 manages the operation of encoding and packetizing unit 122 and attempts to service the demands of terminals 108 in distribution nodes 106. For a particular demand-cast, session manager 112 receives a message from a terminal 108 requesting an IPG page (e.g., via a back-channel), determines whether the requested

page is currently transmitted or available, and directs one of the encoding and packetizing units 122 to encode the requested IPG page and provide the resultant stream(s) to the transport stream generator serving the neighborhood where the requesting terminal resides. Depending on the particular scheme being implemented for the demand-cast (e.g., for a scheme that continually transmit the requested IPG page, as described in Exhibit B), session manager 112 may maintain track of the IPG page being demand-casted so that the page can later be tear down if not needed.

Bandwidth manager 114 assists session manager 112 in managing encoding and packetizing units 122. Bandwidth manager 114 monitors resources usage and availability for encoding and packetizing units 122 and reports this information to session manager 112, which uses the information to efficiently manage the encoding and packetizing units. For demand-cast, bandwidth manager 114 determines whether sufficient bandwidth and PIDs are available in the current transport stream(s) being transmitted to a particular distribution node. Bandwidth manager 114 is aware of the capacity of each transport stream generator 130, e.g., in terms of the number of streams and PIDs that the transport stream generator can support. If the number of (guide, video, audio, and data) streams to be provided to the transport stream generator or the number of PIDs to be used is greater than the capacity of the currently active transport stream(s), bandwidth manager 114 can signal session manager 112 and/or transport stream generator 130 accordingly. Transport stream generator 130 can then generate another transport stream for the additional (guide, video, audio, and data) streams. In an embodiment, bandwidth manager 114 keeps track of the available bandwidth for served streams via correspondences with session manager 112, and session manager 112 keeps track of the PIDs in use. In another embodiment, session manager 112 performs the functions of bandwidth manager 114 and keeps track of the available bandwidth as well as the PIDs in use.

The use of a bank of encoding and packetizing units 120 in head-end 102a can provide enhanced flexibility in generating the transport streams and may also reduce the amount of redundancy in the encoding process. In this embodiment, each IPG page to be transmitted from head-end 102a can be encoded once by any one of the available encoding and packetizing units 122. Each transport stream generator 130 receives the needed streams from selected ones of encoding and packetizing units 122 and generates the required final transport stream(s). In this manner, any distribution node can have access to any IPG page encoded by any encoding and packetizing unit 122.

Each encoding and packetizing unit 122 can be designed to encode the received contents based on a slice-based encoding scheme, as described in Exhibit A, which can provide improved utilization of the available bandwidth. Alternatively or additionally, encoding and packetizing unit 122 can be designed to implement picture-level encoding. Slice and picture-based encoding schemes are described in Exhibit A and in U.S. Patent Application Serial No. 09/384,394, entitled "METHOD AND APPARATUS FOR COMPRESSING VIDEO SEQUENCES," filed April 15, 1999, assigned to the assignee of the invention, and incorporated herein by reference.

In FIG. 1A, the transport stream generators are shown as being located within the head-end and operated to provide the output streams required by the neighborhood being served by the transport stream generators. In some distribution system 100, local neighborhood equipment can be provided to receive one or more transport streams from the head-end, extracts the information (e.g., guide and video slices) in the received transport streams, and combines the extracted information in an order such that the decoder at the terminals can easily decode the IPG without further re-organization. Thus, local neighborhood equipment may include a unit equivalent to the transport stream generator, which is used to generate one or more transport streams required by the neighborhood being served by the local neighborhood equipment.

FIG. 1B is a block diagram of another embodiment of an information distribution system 100b that can also be used to provide interactive program guide and implement the multicasting of the invention. Distribution system 100b includes a head-end 102b, a number of distribution nodes 106a through 106m, and a number of set top terminals 108 coupled to each distribution node.

In the embodiment shown in FIG. 1B, head-end 102b includes session manager 112, bandwidth manager 114, and a bank of transport stream generators 124. Head-end 102b receives contents from content sources 150, with the contents being provided to the bank of transport stream generators 124. Each transport stream generator 126 can be implemented with one or more encoding and packetizing units, such as the ones shown in FIG. 1A and described in Exhibit A. Each transport stream generator 126 receives the appropriate contents (e.g., the IPG pages to be provided on its output transport stream) and generates one or more transport streams. Each transport stream generator 126 also combines the one or more generated transport streams to generate a respective output stream. The output stream is then provided to a distribution node 106 being serviced by that transport stream generator 126.

Session manager 112 manages the operation of transport stream generators 126 and attempts to service the demands of terminals 108 in distribution nodes 106. Session manager 112 receives messages from terminals 108 requesting IPG pages, determines whether the requested pages are currently transmitted or available, and directs the proper transport stream generators 126 to provide the requested IPG pages on the transport streams to be transmitted to the neighborhood where the requesting terminals reside. Again, session manager 112 may maintain track of the IPG pages being demand-casted.

Bandwidth manager 112 can assist session manager 112 in managing transport stream generators 126. Bandwidth manager 114 may monitor resources usage and availability for the transport stream generators 126 and can report this information to session manager 112, which uses the information to efficiently manage the transport stream generators.

Various multiplexing structures and stream indexing schemes can be used to implement the multicast of the invention. The transport stream(s) to be provided to each distribution node can be organized to maximize the number of PID transitions within a transport stream, and to minimize the number of PID transitions between transport streams. Such transport stream organization would facilitate PID transitions and provide improved performance at the terminals because transitions within a transport stream are typically simpler and faster than transitions between transport streams. Some of these multiplexing structures and stream indexing schemes are described below and, for clarity, are described for a specific example in which 40 IPG pages are continually transmitted. These exemplary 40 IPG pages include guide listings for 200 channels in the current and near look-ahead time slots, and each set of ten IPG pages are encoded by their guide and video portions as shown in FIG. 10C in Exhibit A.

FIG. 2A is a diagram of a first multiplexing structure wherein the IPG pages are provided within a single transport stream. For the 40 continually broadcast IPG pages, 40 guide portions can be coded and assigned guide PIDs 1 through 40 and one video portion can be coded and assigned a video PID. An audio input can also be coded and assigned an audio PID, and data can be coded and assigned one or more data PIDs. These guide, video, audio, and data PIDs can be transmitted on a single transport stream 202, as shown in FIG. 2A. With a single transport stream, the terminal is able to retrieve all IPG pages quickly without having to switch between transport streams.

FIG. 2B is a diagram of a second multiplexing structure wherein the IPG pages are provided within multiple transport streams. As noted above, multiple transport streams may be used to provide additional transmission capacity and/or to accommodate more PIDs. In this specific example, the first transport stream 204a includes the first ten IPG pages, the second transport stream 204b includes the second ten IPG pages, the third transport stream 204c includes the next ten IPG pages, and the fourth transport stream 204d includes the last ten IPG pages. Different number of transport streams can also be used to send the IPG depending on various factors such as, for example, bandwidth usage, demands, and so on. Also, each transport stream can include different number of IPG pages than that shown in FIG. 2B.

As shown in FIG. 2B, the 40 IPG pages to be transmitted are multiplexed into four sets, one set of IPG pages for each transport stream. Each set can be defined to include guide listings for successive channels. For example, the first set can include guide listings for channels 1 through 100 for the current time slot, the second set can include guide listings for channels 101 through 200 for the current time slot, the third set can include guide listings for channels 1 through 100 for the next near look-ahead time slot, and the last set can include guide listings for channels 101 through 200 for the next near look-ahead time slot. This multiplexing structure may reduce the number of transitions between transport streams when scrolling through the guide listings.

As described above, when a viewer selects a new IPG page for viewing, a determination is first made at the terminal whether the selected IPG page is transmitted in the current transport stream(s). If the answer is yes, the guide PID for the selected page is extracted from the transport stream, decoded, and presented. Otherwise, if the selected IPG page is transmitted on another transport stream, the terminal first recovers that transport stream and then processes the selected guide PID. Additional processing time is typically required to switch between transport streams, which may result in noticeable delays at the terminal.

FIG. 2C is a diagram of a third multiplexing structure wherein the IPG pages are provided within multiple transport streams with overlapping guide PIDs. To reduce processing delays associated with switching between transport streams at the terminal (e.g., when a viewer is scrolling through program listings), some of the IPG pages can be transmitted on multiple transport streams. In the overlapping multiplexing structure, the IPG pages to be transmitted are also partitioned into four sets of IPG pages, one set for each transport stream. However, each set includes one or more IPG pages

from a neighboring set. In the example shown in FIG. 2C, the first transport stream includes IPG pages 1 through 11, the second transport stream includes IPG pages 11 through 21, the third transport stream includes IPG pages 21 through 31, and the fourth transport stream includes IPG pages 31 through 40.

With the overlapping multiplexing structure, when a viewer tunes to an "end" IPG page, the terminal can initiate the processing of the adjacent transport stream so that this transport stream is available if the viewer selects an IPG page from this transport stream. For example, when the viewer tunes to IPG page 11 in the first transport stream, the terminal can initiate the processing of the second transport stream so that this transport stream is available if the viewer selects IPG page 12. The amount of overlapping between transport streams (e.g., one page in FIG. 2C) can be dependent on various factors such as, for example, the time required to acquire a new transport stream, the available bandwidth in the transport streams, and so on.

To also reduce processing delays, a frequently accessed IPG page can be transmitted in multiple transport streams. For example, one or more IPG pages in the current time period can be transmitted in each transport stream being provided to a neighborhood. As another example, a popular IPG page (e.g., for a big sporting event) can be transmitted in a number of transport streams. Inclusion of the frequently accessed IPG page in multiple transport streams allows a terminal processing any one of these transport streams to be able to retrieve the IPG page more quickly without having to switch to another transport stream.

Referring back to FIG. 1A, each IPG page can be coded by one of the available encoding and packetizing units 122. The resultant guide stream from the encoding and packetizing unit 122 can then be sent to one or more transport stream generators 130. Each transport stream generator 130 can be directed (e.g., by session manager 112) to include the guide stream in one or more transport streams. In this manner, various multiplexing structures can be implemented without additional burden on the encoding units.

The particular multiplexing structure for a transport stream generator can also be selected based on various factors associated with that transport stream generator. For example, the multiplexing structure may be selected based on the available bandwidth, the demands, and other factors. The multiplexing structure can also be adjusted dynamically. For example, more overlapping may be utilized if more bandwidth is available or as more transport streams are added. Also, each transport stream generator

within the head-end may be operated to implement a particular multiplexing structure especially suited for the neighborhood being served by that transport stream generator.

For clarity, the multicasting, multiplexing structure, and stream indexing schemes of the invention have been specifically described for the delivery of IPG. However, these techniques can also be adapted for delivery of other services and contents. In general, any content (including IPG) can be coded, assigned respective PIDs, multiplexed into one or more transport streams, and transmitted from the head-end to a neighborhood. For example, the techniques of the invention can be used to deliver stock quotes, sports scores, headline news, traffic reports, other guides, and so on. During periods of heavy demands (e.g., during prime time), more transport streams can be transmitted to meet increased demands. The transport streams can subsequently be tear down when not needed.

The foregoing description of the preferred embodiments is provided to enable any person skilled in the art to make or use the invention. Various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments without the use of the inventive faculty. Thus, the invention is not intended to be limited to the embodiments shown herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

WHAT IS CLAIMED IS:

1. A system for providing interactive program guide (IPG), the system comprising:

a plurality of encoding units operative to encode a plurality of IPG pages and generate a plurality of streams, wherein each stream is assigned a respective packet identifier (PID);

at least one transport stream generator operatively coupled to the plurality of encoding units, each transport stream generator operative to receive and multiplex selected ones of the plurality of streams from one or more encoding units into one or more transport streams; and

a session manager coupled to the at least one transport stream generator and operative to direct each transport stream generator to generate the one or more transport streams based on usage.

2. The system of claim 1, further comprising:

a bandwidth manager coupled to the plurality of encoding units and the session manager, the bandwidth manager operative to monitor usage and report to the session manager.

3. The system of claim 1, wherein the plurality of encoding units are operative to encode only once each IPG page to be transmitted from the at least one transport stream generator.

4. The system of claim 1, wherein the number of transport streams generated by each transport stream generator is dynamically adjusted based on demands from a neighborhood being served by the transport stream generator.

5. The system of claim 1, wherein the session manager directs a particular transport stream generator to generate an additional transport stream as usage increases and exceeds the capacity of currently transmitted transport stream(s).

6. The system of claim 1, wherein the session manager directs a particular transport stream generator to generate an additional transport stream if the number of

streams to be transmitted by the particular transport stream generator exceeds the capacity of currently transmitted transport stream(s).

7. The system of claim 1, wherein the session manager directs a particular transport stream generator to generate an additional transport stream if a required number of PIDs exceeds a maximum number of PIDs supported by currently transmitted transport stream(s).

8. The system of claim 1, wherein the session manager directs a particular transport stream generator to tear down a transport stream if usage falls below the capacity of remaining transport streams.

9. The system of claim 1, wherein each transport stream generator is operative to serve a respective group of terminals within a particular neighborhood.

10. The system of claim 1, wherein each transport stream generator is operable to provide differentiated IPG via the one or more transport streams generated by the transport stream generator.

11. The system of claim 1, wherein a plurality of transport streams are generated by a particular transport stream generator, and wherein each of the plurality of transport streams includes a respective set of IPG pages.

12. The system of claim 11, wherein the plurality of transport streams from the particular transport stream generator include overlapping sets of IPG pages.

13. The system of claim 11, wherein each of the plurality of transport streams from the particular transport stream generator includes one or more common IPG pages.

14. The system of claim 11, wherein the sets of IPG pages for the plurality of transport streams from the particular transport stream generator are organized to reduce likelihood of switching between transport streams at a receiving device.

15. The system of claim 11, wherein the sets of IPG pages for the plurality of transport streams from the particular transport stream generator are organized to increase likelihood of PID transitions within the same transport stream.

16. The system of claim 1, wherein each encoding unit is operative to implement a slice-based encoding scheme.

17. The system of claim 1, wherein each encoding unit is operative to implement a picture-based encoding scheme.

18. A system for providing interactive program guide (IPG), the system comprising:

at least one transport stream generator, each transport stream generator including at least one encoder unit operative to encode a plurality of IPG pages and generate a plurality of streams, each transport stream generator operative to generate one or more transport streams having included therein the plurality of streams generated for the plurality of encoded IPG pages;

a session manager coupled to the at least one transport stream generator and operative to direct each transport stream generator to generate the one or more transport streams based on usage.

19. The system of claim 18, wherein each of the plurality of streams generated for the plurality of IPG pages is assigned a respective packet identifier (PID).

20. A method for providing interactive program guide (IPG) from a transmission source to a plurality of terminals, the method comprising:

monitoring demands from the plurality of terminals;

determining a current capacity of one or more transport streams being transmitted for the plurality of terminals;

comparing the demands from the plurality of terminals against the current capacity; and

dynamically adjusting the number of transport streams to be transmitted to the plurality of terminals based on a result of the comparing.

21. The method of claim 20, further comprising:

providing an additional transport stream for the plurality of terminals if the demands exceeds the current capacity.

22. The method of claim 20, further comprising:

providing an additional transport stream for the plurality of terminals if a required number of packet identifiers (PIDs) exceeds a maximum number of PIDs supported by the one or more transport streams currently being transmitted.

23. The method of claim 20, further comprising:

tearing down a particular currently transmitted transport stream if the demands fall below the capacity of remaining transport streams.

METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS

ABSTRACT OF THE DISCLOSURE

A system for delivering interactive program guide (IPG) includes a number of encoding units, at least one transport stream generator, and a session manager. The encoding units encode a number of IPG pages and generate a number of (e.g., guide, video, audio, and data) streams, with each stream being assigned a respective packet identifier (PID). Each transport stream generator receives and multiplexes selected ones of the streams from one or more encoding units into one or more transport streams. The session manager directs each transport stream generator to generate one or more transport streams based on usage. The system may further include a bandwidth manager that monitors usage and reports the usage to the session manager. The encoding units can be operated to encode only once each IPG page to be transmitted. Each transport stream generator can be operated to provide differentiated IPG for the neighborhood being served by the transport stream generator. The number of transport streams generated by each transport stream generator can be dynamically adjusted based on demands from the neighborhood being served by the transport stream generator. Each transport stream generator can be directed to generate an additional transport stream if usage exceeds the capacity of the currently transmitted transport streams.

EXHIBIT A

Application Serial No. 09/466,990

STREAM INDEXING FOR DELIVERY OF INTERACTIVE PROGRAM GUIDE

CROSS-REFERENCES TO RELATED APPLICATIONS

This application claims benefit of U.S. Provisional Patent Application Serial No. 60/129,598 filed April 15, 1999.

This application is also a continuation-in-part of U.S. Patent Application Serial No. 09/293,535, entitled "IMPROVED DATA STRUCTURE AND METHODS FOR PROVIDING AN INTERACTIVE PROGRAM GUIDE", filed April 15, 1999.

This application is also a continuation-in-part of U.S. Patent Application Serial No. 09/384,394, entitled "METHOD AND APPARATUS FOR COMPRESSING VIDEO SEQUENCES," filed August 27, 1999.

This application is also a continuation-in-part of U.S. Patent Application Serial No. 09/428,066, entitled "METHOD AND APPARATUS FOR TRANSMITTING VIDEO AND GRAPHICS IN A COMPRESSED FORM," filed October 27, 1999.

The above-identified related applications are all assigned to the assignee of the present invention and incorporated herein by reference in their entirety for all purposes.

BACKGROUND OF THE INVENTION

The present invention relates to communications systems in general. More specifically, the invention relates to techniques to efficiently deliver interactive program guide (IPG) in a server-centric system.

Over the past few years, the television industry has seen a transformation in a variety of techniques by which its programming is distributed to consumers. Cable television systems are doubling or even tripling system bandwidth with the migration to hybrid fiber coax (HFC) cable plant. Customers unwilling to subscribe to local cable systems have switched in high numbers to direct broadcast satellite (DBS) systems. And, a variety of other approaches have been attempted focusing primarily on high bandwidth digital technologies, intelligent two way set top terminals, or other methods of trying to offer service differentiated from standard cable and over the air broadcast systems.

With this increase in bandwidth, the number of programming choices has also increased. Leveraging off the availability of more intelligent set top terminals, several companies such as Starsight Telecast Inc. and TV Guide, Inc. have developed

elaborate systems for providing an interactive listing of a vast array of channel offerings, expanded textual information about individual programs, and the ability to look forward to plan television viewing as much as several weeks in advance.

With this increase in the quantity of programming, it is a challenge to deliver program guide data to viewers in an efficient and effective manner. A large amount of resources (e.g., bandwidth) would normally be needed to continually transmit, for example, two weeks of programming for 200 channels. Therefore, efficient and effective techniques to provide interactive program guide to a large number of viewers are highly desirable.

SUMMARY OF THE INVENTION

The present invention is directed to stream indexing for delivery of an interactive program guide. These techniques overcome the above described problems and disadvantages.

In accordance with a first aspect of the present invention, a method of stream indexing for delivery of an interactive program guide comprises: assigning a first plurality of packet identifiers to program guide content for a current time period; and assigning a second plurality of packet identifiers to program guide content for a plurality of look-ahead time periods.

In accordance with a second aspect of the present invention, a method of stream indexing for delivery of an interactive program guide (IPG) comprises: providing a plurality of video packet identifiers; assigning each video packet identifier to a corresponding guide page; providing a plurality of data packet identifiers, where the plurality of data packet identifiers is less in number than the plurality of video packet identifiers; predetermining a prime number which is less in number than or equal in number to the plurality of video packet identifiers; dividing each video packet identifier by the prime number in order to generate a remainder; and using the remainder to assign a data packet identifier to each video packet identifier.

The foregoing, together with other aspects of this invention, will become more apparent when referring to the following specification, claims, and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings.

FIG. 1 is a block diagram of an embodiment of an interactive information distribution system that can implement various aspects of the present invention;

FIG. 2 is a block diagram of an encoding and multiplexing unit in accordance with an embodiment of the present invention;

FIG. 3 is a flow diagram of a process used by a picture isolator within the encoding and multiplexing unit;

FIG. 4 is a data structure of a transport stream is generated by a head-end;

FIG. 5 is a block diagram of a receiver within a subscriber equipment suitable for use in the interactive information distribution system;

FIGS. 6-8 are flow diagrams of the first, second, and third methods, respectively, for recombining and decoding streams;

FIG. 9 is an example of one picture taken from a video sequence that can be encoded using the invention;

FIGS. 10A-10C are matrix representations of program guide data with various data groupings for efficient encoding in accordance with the invention;

FIG. 11 is a diagram of an embodiment of a slice division of an IPG page;

FIG. 12A is a diagram of a server-centric system architecture for managing delivery of an interactive user interface;

FIG. 12B is a diagram of a line neighborhood equipment;

FIG. 13 is a flow diagram of a process for generating a portion of transport stream containing intra-coded video and graphics slices;

FIGS. 14 and 15 are flow diagrams of two processes for generating a portion of transport stream containing predictive-coded video and graphics slices;

FIG. 16 is a diagram of a data structure of a transport stream used to transmit the IPG page shown in FIG. 9;

FIGS. 17A and 17B are diagrams of an IPG page having a graphics portion and a number of video portions and a corresponding slice map for the IPG page, respectively;

FIG. 18 is a flow diagram of a process for generating a portion of transport stream containing intra-coded video and graphics slices for an IPG having a graphics portion and a number of video portions;

FIG. 19 is a flow diagram of a process for generating a portion of transport stream containing predictive-coded video and graphics slices for an IPG having a graphics portion and a number of video portions;

FIG. 20 is a block diagram illustrating an apparatus for encoding, packetizing, multiplexing, and assigning programs to video, audio, and data in accordance with a "level zero" embodiment of the invention;

FIGS. 21A and 21B are diagrams illustrating a program assignment structure for a multiple-program final transport stream and a single-program final transport stream, respectively, in accordance with a "level zero" embodiment of the invention;

FIG. 22 is a diagram illustrating multiplexing of video, audio, and data packets into a final transport stream in accordance with a "level zero" embodiment of the invention;

FIG. 23 is a diagram illustrating an assignment structure for multiple final transport streams in accordance with a "level zero" embodiment of the invention;

FIG. 24 is a diagram illustrating a final transport stream in accordance with a "level one" embodiment of the invention;

FIGS. 25A and 25B are diagrams illustrating multiple final transport streams in accordance with a "level one" embodiment of the invention;

FIG. 26 is a diagram illustrating a final transport stream in accordance with a "level two" embodiment of the invention;

FIG. 27A is a diagram illustrating a technique for reducing switching latencies by carrying redundant packets in accordance with an embodiment of the invention;

FIG. 27B is a diagram illustrating slice-based multiple transport streams with overlapping PIDs to reduce latencies in accordance with an embodiment of the invention;

FIG. 28 is a diagram illustrating an IPG page with two threshold levels for stream priming in accordance with an embodiment of the invention;

FIG. 29 is a diagram illustrating a program mapping table (PMT) in accordance with an embodiment of the invention;

FIGS. 30A and 30B are diagrams illustrating prime time slots and half-hour shifts of a current programming time slot, respectively, in accordance with an embodiment of the invention;

FIG. 31 is a diagram illustrating a mapping of look-ahead video PIDs to look-ahead data PIDs in accordance with an embodiment of the invention;

FIG. 32 is a diagram illustrating television usage time during a typical week;

FIGS. 33A and 33B are diagrams illustrating a first look-ahead video PID layout and a method of forming a second look-ahead video PID layout, respectively, in accordance with an embodiment of the invention;

FIG. 33C is a diagram illustrating the distribution of data messages among data PIDs in accordance with an embodiment of the invention;

FIG. 34 is a block diagram of a receiver within subscriber equipment suitable for use in an interactive information distribution system; and

FIGS. 35-38 are flow diagrams of the first, second, third, and fourth slice recombination processes, respectively, in accordance with an embodiment of the invention.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common within a figure.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

PICTURE-LEVEL PROCESSING

A. SYSTEM

FIG. 1 is a high-level block diagram of an information distribution system 100 (e.g., a video-on-demand system or digital cable system) that can be used to implement various aspects of the invention. System 100 includes a head-end 102 (e.g., a service provider equipment), a distribution network 106 (e.g., hybrid fiber-coax network), and a number of terminals 108. This architecture of information distribution system is disclosed in commonly assigned U.S. Patent Application Serial No. 08/984,710, filed

December 3, 1997. One implementation of system 100 is a DIVA™ system provided by DIVA Systems Corporation.

Head-end 102 produces a number of digital streams that contain encoded information in (e.g., MPEG) compressed format. These streams are modulated using a modulation format that is compatible with distribution network 106. Terminals 108a through 108n are located at various subscriber locations. Upon receiving a stream, terminal 108 extracts the information from the received signal and decodes the stream to produce a signal containing various contents (e.g., produce a television program, program guide page, or other multimedia program) suitable for a display unit.

In an interactive information distribution system such as the one described in the aforementioned U.S. Patent Application Serial No. 08/984,710, the program streams are addressed to the particular terminals that requested the information through an interactive menu. Interactive menu structures for requesting video on demand are disclosed in commonly assigned U.S. Patent Application Serial No. 08/984,427, filed December 3, 1997 and Serial No. 60/093,891, filed in July 23, 1998.

To assist a viewer in selecting programming, head-end 102 produces an interactive program guide (IPG) that is compressed for transmission in accordance with the invention. The IPG contains program information (e.g., title, time, channel, program duration and the like) as well at least one region displaying full motion video (e.g., a television advertisement or promotion). Such informational video is provided in various locations within the program guide screen.

The invention produces the IPG using a compositing technique that is described in commonly assigned U.S. Patent Applications Serial No. 09/201,528, filed November 30, 1998 and [Attorney Docket No. 19880-000500], filed July 23, 1999, which are hereby incorporated by reference herein. The compositing technique, which is not described herein, enables full motion video to be positioned within an IPG and allows the video to seamlessly transition from one IPG page to another. The composited IPG pages (i.e., a number of video frame sequences) are coupled from a video source 112 to an encoding and multiplexing unit 116. One or more audio signals associated with the video sequences are also supplied by an audio source 114 to encoding and multiplexing unit 116.

Encoding and multiplexing unit 116 compresses the frame sequences into a number of elementary streams, which are further processed to remove redundant

information. A multiplexer within unit 116 then assembles the elementary streams into one or more transport streams.

Each transport stream is then modulated by a digital video modulator 122 based on a modulation format that is compatible with distribution network 106. For example, in the DIVA™ system, the modulation is quadrature amplitude modulation (QAM). However, other modulation formats can also be used.

Each terminal 108 includes a receiver and a display (e.g., a television). The receiver demodulates the signals carried by distribution network 106 and decodes the demodulated signals to extract the IPG pages from the stream. An design of terminal 108 is described in further detail below.

1. Encoding and Multiplexing Unit

FIG. 2 is a block diagram of an embodiment of encoding and multiplexing unit 116, which can be used to produce one or more transport streams comprising a number of encoded video, audio, and data elementary streams. Encoding and multiplexing unit 116 can be advantageously used in an ensemble encoding environment, whereby a number of video streams are generated to compress video information that carries common and non-common content. In an embodiment, the common content is encoded into a single elementary stream and the non-common content is encoded into separate elementary streams. In this way, the common content is not duplicated in every stream, which can yield significant bandwidth savings. In a practical MPEG encoding process, some common information will likely appear in the stream intended to carry non-common information and some non-common information will likely appear in the stream intended to carry common information.

Although the following description is presented within the context of IPG, the method and apparatus described herein can be applied to a broad range of applications, such as broadcast video on demand delivery, e-commerce, Internet, video education services, and others. The method and apparatus described can be advantageously used to deliver video sequences with command content.

In the embodiment shown in FIG. 2, encoding and multiplexing unit 116 receives a number of video sequences (e.g., V1 through V10) and, optionally, one or more audio signals and one or more data streams (only one audio signal and one data stream in shown in FIG. 2). The video sequences V1-V10 include imagery common to

each other (e.g., common IPG background information and common video portion information). Each video sequence further includes imagery specific to the sequence (e.g., the programming information, program grid graphic) and different from those of other sequences.

The audio signal(s) comprises audio information that may be associated with a video portion in the video sequences (e.g., an audio track associated with still or moving images). For example, if video sequence V1 represents a movie trailer, the audio signal can be derived from an audio source (e.g., music and voice-over) associated with the music trailer.

The data stream can comprises overlay graphics information, textual information describing programming indicated by the guide region, and other system or user interface related data. The data stream can be separately encoded into its own elementary stream or included within the (e.g., MPEG-2) transport stream. The data stream can be suitable for use in the information distribution system as private data, auxiliary data, and the like.

In the embodiment shown in FIG. 2, encoding and multiplexing unit 116 includes an encoding profile and clock generator 202, a number of real-time video (e.g., MPEG-2) encoders (RTE) 220a through 220j, an audio delay element 222, a real-time audio (e.g., AC-3) encoder 224, an optional data processor 226, a number of picture isolators 230a through 230j, a number of packetizers 240a through 240m, a number of buffers 250a through 250m, and a transport multiplexer 260.

The video sequences V1-V10 are coupled to respective real-time encoders 220. Each encoder 220 encodes, illustratively, a composited IPG screen sequence to form a corresponding compressed video bit stream, e.g., an MPEG-2 compliant bit stream having associated with it a particular group of pictures (GOP) structure. The common clock and encoding profile generator 202 provides a clock and profile to each encoder 220 to ensure that the encoding timing and encoding process occur similarly for each video sequence V1-V10. This allows the video sequences to be encoded in a synchronous manner.

For the following description, it is assumed that the GOP structure consists of an I-picture followed by ten B-pictures, with a P-picture separating each group of two B-pictures (i.e., "I-B-B-P-B-B-P-B-B-P-B-B-P-B-B"). However, any GOP structure and size may be used in different configurations and applications. It is preferable that the

same encoding profile, including the GOP structure, is used by each of real time encoders 220 to have uniform encoding across multiple streams and to produce approximately the same size encoded I and predicted pictures. Moreover, by utilizing the same profile and predefined GOP structure, multiple instances of the same encoder can be used to implement encoding and multiplexing unit 116, which can reduce implementation costs. It can be noted also that the encoding process can be performed by one or a number of encoders depending on the particular implementation.

Each real time encoder 220 produces an encoded (e.g., MPEG-2 compliant) bit stream that is coupled to a respective picture isolator 230. Each picture isolator 230 examines the encoded video stream (E) to isolate the I pictures within the bit streams by analyzing the stream access units associated with the I, P, and B pictures.

Picture isolators 230 process the received streams E1-E10 according to the type of picture (I, P or B picture) associated with a particular access unit (described below) and also the relative position of the pictures within the sequence and group of pictures. The first picture isolator 230a receives the bit stream E1 from the first real time encoder 220a and, in response, produces two output bit streams PRED and I1. The remaining picture isolators 230b to 230j produce only I-picture streams. It can be noted that the PRED stream can be generated by any one of the picture isolators.

As noted in the MPEG-1 and MPEG-2 specifications, an access unit comprises a coded representation of a presentation unit. In the case of audio, an access unit is the coded representation of an audio frame. In the case of video, an access unit includes all the coded data for a picture and any stuffing bits that follows it, up to but not including the start of the next access unit. If a picture is not preceded by a group start code or a sequence header code, then the corresponding access unit begins with the picture start code. If the picture is preceded by a group start code and/or a sequence header code (e.g., for an I-picture), then the corresponding access unit begins with the first byte of the first start code in the sequence or a GOP. If the picture is the last picture preceding a sequence end code in the stream, then all bytes between the last byte of the coded picture and the sequence end code (including the sequence end code) belong to the access unit. Each B and P-picture access unit in a GOP includes a picture start code. The last access unit of the GOP (e.g., a terminating B-picture) includes, in addition, a sequence end code indicating the termination of the GOP.

The I1 stream, as the first picture of the sequence, comprises a sequence header, a sequence extension, a GOP header, a picture header, a picture extension, and the I-picture data until the next picture start code. The PRED stream comprises only P and B picture access units, starting from the second picture start code (illustratively a B-picture) and all data until the next group start code. Thus, the PRED stream includes all access units of the GOP except those representing the I-picture.

The remaining picture isolators 230b through 230j respectively receive the (e.g., MPEG-2 compliant) streams E2 through E10 from the corresponding real-time encoders 220b through 220j and respectively produced the output stream I2 through I10. Each output stream comprises only the sequence header and all data until the second picture start codes (i.e., the access unit data associated with an I-picture at the beginning of the respective GOP).

FIG. 3 is a flow diagram of an embodiment of a process 300 for isolating pictures, which is suitable for use with picture isolators 230 in FIG. 2. At step 310, the picture isolator waits for a sequence header or a group start code. Upon detecting this, the sequence header and all data until the second picture start code is accepted, at step 315. The accepted data is then coupled to the I-picture output of the picture isolator, at step 320. For picture isolators 230b through 230j, since there are no predicted pictures output, the accepted data (i.e., the sequence header, I-picture start code and I-picture) is coupled to a single output.

At step 325, a query is made whether non-I-picture data is to be processed (i.e., discarded or coupled to a packetizer). If the non-I-picture data is to be discarded, then the process returns to step 310 to wait for the next sequence header. Otherwise, if the non-I-picture data is to be coupled to a packetizer, the second picture start code and all data in a GOP until the next group start code is accepted, at step 330. The accepted data is then coupled to the non-I-picture output of frame isolator 230 to form the PRED stream, at step 335.

Thus, picture isolator examines the compressed video stream produced by real time encoder 220 to identify the start of a GOP, the start of an I-picture (i.e., the first picture start code after the group start code), and the start of the predicted pictures (i.e., the second picture start code after the group start code) forming the remainder of a GOP. The picture isolator couples the I-pictures and predicted pictures to the packetizers for further processing.

The first packetizer 240a packetizes the PRED stream into a number of fixed length transport packets according to, for example, the MPEG-2 standard. Additionally, the first packetizer 240a assigns a packet identification (PID) (e.g., PID1) to each of the packets including information from the PRED stream, thereby producing a packetized stream PID1. The second packetizer 240b packetizes the I stream to produce a corresponding packetized stream PID2. The I2 through I10 output streams of the second through tenth picture isolators 230b through 230j are respectively coupled to the third through eleventh transport packetizers 240c through 240k, which respective produce the packetized streams PID3 through PID11.

In addition to the video information forming the ten IPG pages, audio information associated with IPG pages is encoded and supplied to transport multiplexer 260. Specifically, the audio signal is provided to audio delay 222 and then encoded by a real-time audio encoder 224 (e.g., a Dolby AC-3 real-time encoder) to produce an encoded audio stream. The encoded stream is then packetized by the 12th transport packetizer 240l to produce a transport stream assigned with a particular PID (e.g., PID12). The packetized audio stream with PID12 is coupled to the 12th buffer 250l.

In an embodiment, the IPG grid foreground and overlay graphics data is coupled to transport multiplexer 260 as a coded data stream assigned with a particular PID (e.g., PID13). The coded data stream is produced by processing the input data stream as related for the application using data processor 226 and packetizing the processed data stream using the thirteenth packetizer 240m to produce the packetized data stream with PID13, which is coupled to the thirteenth buffer 250m.

The packetized streams from packetizers 240a through 240k are respectively coupled to buffer 250a through 250k, which are in turn coupled to respective inputs of multiplexer 260. In an embodiment, multiplexer 260 is an MPEG-2 transport multiplexer. While any type of multiplexer can be used to practice the invention, various aspects of the invention are described within the context of an MPEG-2 transport multiplexing system.

As defined in the MPEG-2 specification (formally referred to as the ISO standard 13818-1), a transport stream is a sequence of equal sized packets, with each packet being 188 bytes in length. Each packet includes a 4-byte header and 184 bytes of data. The header contains a number of fields, including a 13-bit PID field that uniquely identifies each packet that contains a portion of a "stream" of video information as well as

audio information and data. As such, to decode a particular video stream (or audio or data stream) for viewing or presentation, the decoder in the terminal extracts packets containing a particular PID and decodes those packets to create the video (or audio or data) for viewing or presenting.

In an embodiment, each of the thirteen streams representing a portion of the IPG is uniquely identified by a PID. In an embodiment, the thirteen streams are multiplexed into a single transport stream. Fewer or more IPG streams may be included in the transport stream as bandwidth permits. Additionally, more than one transport stream can be used to transmit the IPG streams.

Multiplexer 260 processes the packetized data stored in each of the 13 buffers 250a through 250m in a particular order (e.g., in a round robin basis, beginning with the 13th buffer 250m and concluding with the first buffer 250a). For the round robin order, transport multiplexer 260 retrieves or "drains" the packetized data stored within the 13th buffer 250m and couples that data to the output stream Tout. Next, the 12th buffer 2501 is emptied and the packetized data stored therein is coupled to the output stream Tout. Next, the 11th buffer 250k is emptied and the packetized data stored therein which is coupled to the output stream Tout. The process continues until the 1st buffer 250a is emptied and the packetized data stored therein is coupled to the output stream Tout. The processing flow can be synchronized such that each output buffer includes all the access units associated with an I-picture (250b through 250k) suitable for referencing a GOP, a particular group of P and B pictures (250a) suitable for filling out the rest of the GOP, a particular one or more audio access units (2501), and a related amount of data (250m). The round robin draining process is repeated for each buffer, which has been filled in the interim by new transport packetized streams.

FIG. 4 depicts a transport stream 400 produced by encoding and multiplexing unit 116 as a result of processing the input streams in a round robin basis. FIG. 4 shows one GOP portion of the transport stream, which is indicated by the "START" and "END" phrases. The GOP starts with data packet 401 assigned with PID13, then an audio packet 402 assigned with PID12, which are followed by I-picture packets 403 through 412 assigned as PID11 through PID-2. The remaining packets 413 through 425 carry the PRED stream with PID1. Packets 423 to 425 in FIG. 4 show the terminating access units of the previous GOP.

Note that the exemplary transport stream and the round robin process are not required for the operation of the invention. The data and audio packets can be placed into different parts of the transport stream, or the sequence of I-picture packets can be provided in a different order. To allow the terminal to decode the transport stream in one pass without storing any packets, the packets for the I-pictures should precede the packets for the PRED pictures in the transport stream. This output order is needed since the reference I-pictures need to be decoded before the predicted pictures. However, a different order can be used if the terminals have the necessary storage capabilities.

In an embodiment, the IPG streams are encapsulated in one multi-program transport stream. Each program in the program map table (PMT) of an MPEG-2 transport stream includes an I-PID (one of the illustrative ten I-PIDs 403 to 412), the PRED stream PID1, data PID13 401, and audio PID12 402. Although multiplexer 260 of FIG. 2 couples a PRED stream access units 413 to 425 to the multiplexer output Tout only once per GOP, the PMT for each program references the same PRED stream PID1. For the illustrative organization of video inputs in FIG. 2, ten programs can be formed with each program consisting of one of the ten I-PIDs 403 to 413, the PRED PID1, the audio PID12, and the data PID13.

In another embodiment, the information packets are formed into a single program and carried with a single-program transport stream. In this embodiment, the complete set of PIDs 401 to 425 is coupled into a single program. In yet another embodiment, multiple transport streams are employed to send the IPG. In this embodiment, each transport stream can be formed as a single program or as multiple programs, with each program comprising an I-PID, the PRED-PID, the data PID, and the audio PID. The information packets in each transport stream are retrieved in a similar manner as for the single transport stream. In yet another embodiment, the information packets are carried in single program multiple transport streams. Thus, a variety of transport stream formats can be employed to carry the generated streams.

B. RECEIVER

FIG. 5 depicts a block diagram of an embodiment of terminal 108 (also referred to as a set top terminal (STT) or user terminal) suitable for producing a display of a user interface in accordance with the invention. Terminal 108 includes a tuner 512, a demodulator 514, a transport demultiplexer 518, an audio decoder 520, a video decoder

530, an on-screen display (OSD) processor 532, a video compositor 534, a frame store memory 536, a controller 550, and a modulator 570. User interaction is provided via a remote control unit 580. Tuner 512 receives, e.g., a radio frequency (RF) signal comprising, for example, a number of quadrature amplitude modulated (QAM) information signals from a downstream (forward) channel. Tuner 512, in response to a control signal TUNE, tunes to and processes a particular QAM information signal to produce an intermediate frequency (IF) information signal. Demodulator 514 receives and demodulates the IF information signal to produce an information stream, illustratively an MPEG transport stream. The MPEG transport stream is provided to a transport stream demultiplexer 518.

Transport stream demultiplexer 518, in response to a control signal TD produced by controller 550, demultiplexes (i.e., extracts) an audio information stream A and a video information stream V. The audio information stream A is provided to audio decoder 520, which decodes the audio information stream and provides a decoded audio information stream to an audio processor (not shown) for subsequent presentation. The video stream V is provided to video decoder 530, which decodes the compressed video stream V to produce an uncompressed video stream VD that is provided to video compositor 534. OSD processor 532, in response to a control signal OSD produced by controller 550, produces a graphical overlay signal VOSD that is provided to video compositor 534. During transitions between streams representing the user interfaces, the buffers in the decoder are not reset. As such, the user interfaces seamlessly transition from one screen to another.

Video compositor 534 merges the graphical overlay signal VOSD and the uncompressed video stream VD to produce a modified video stream (i.e., the underlying video images with the graphical overlay) that is provided to frame store unit 536. Frame store unit 536 stores the modified video stream on a frame-by-frame basis according to the frame rate of the video stream. Frame store unit 536 provides the stored video frames to a video processor (not shown) for subsequent processing and presentation on a display device.

Controller 550 includes an input/output module 552, a microprocessor 554, support circuitry 556, an infrared (IR) receiver 558, and a memory 560. Input/output module 552 forms an interface between controller 550 and tuner 512, transport demultiplexer 518, OSD processor 532, back-channel modulator 570, and remote control

unit 580. Microprocessor 554 cooperates with support circuitry 556 such as power supplies, clock circuits, cache memory, and the like as well as circuits that assist in executing the software routines that are stored in memory 560.

Although controller 550 is depicted as a general-purpose processor that is programmed to perform specific interactive program guide control function in accordance with the invention, the controller can be implemented in hardware as an application specific integrated circuit (ASIC). As such, the process steps described herein are intended to be broadly interpreted as being equivalently performed by software, hardware, or a combination thereof.

In the embodiment shown in FIG. 5, remote control unit 580 includes an 8-position joystick, a numeric pad, a "Select" key, a "Freeze" key and a "Return" key. User manipulations of the joystick or keys of the remote control device are transmitted to controller 550 via an infrared (IR) link or an RF link. Controller 550 is responsive to such user manipulations, executes related user interaction routines 562, and uses particular overlays that are available in an overlay storage 566.

Once received, the video streams are recombined via stream processing routine 568 to form the video sequences that were originally compressed. The following describes three illustrative methods for recombining the streams.

1. Recombination Method 1

In the first recombination method, the I-picture stream and the predicted picture streams to be recombined keep their separate PIDs until the point where they are depacketized. The recombination process is conducted within the transport demultiplexer of the terminal. For illustrative purposes, in a multi-program transport stream, each program consists of an I-PID for the I-picture, the PRED-PID for the predicted pictures, an audio PID, and a data PID. Any packet with a PID that matches any of the PIDs within the desired program (as identified in a program mapping table) are depacketized and the payload is sent to the video decoder. Payloads are sent to the decoder in the order in which the packets arrive at the demultiplexer.

FIG. 6 is a flow diagram of an embodiment of a first recombination process 600. At step 610, the process waits for a (viewer) selection for a picture (e.g., a particular IPG page) to be received. The I-PID for the selected picture, as the first picture

of a video stream's GOP, identifies the stream to be received. A packet having the identified I-PID is then detected.

At step 615, the I-PID packets are extracted from the transport stream, including the header information and data, until the next picture start code. The header information within the first received I-PID access unit includes a sequence header, a sequence extension, a group start code, a GOP header, a picture header, and a picture extension, which are known to a reader that is skilled in MPEG-1 and MPEG-2 compression standards. The header information in the next I-PID access unit that belongs to the second and later GOPs includes the group start code, the picture start code, the picture header, and an extension. At step 620, the payloads of the packets that include header information related to the video stream and the intra-coded picture are coupled to the video decoder as video information stream V.

At step 625, the predicted picture packets PRED-PID (e.g., PID1 in FIG. 2) for fourteen predictive-coded pictures in a GOP of size fifteen are extracted from the transport stream. At step 630, the payloads of the packets that include the header information related to the video stream and the predicted picture data are coupled to the video decoder as video information stream V. At the end of step 630, a complete GOP, including the I-picture and predicted pictures, are available to the video decoder. As the payloads are sent to the decoder in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination processing.

At step 635, a query is then made whether a different picture is requested, (e.g., a new IPG is selected). If a different picture is not requested, then the process returns to step 610 and the demultiplexer waits for the next packets having the PID of the desired I-PID. Otherwise, if a different picture is requested, then the I-PID of the new desired picture is identified at step 640, and the process returns to step 610.

The process shown in FIG. 6 can be used to produce an MPEG-compliant video stream V by recombining the desired I-picture and the predicted pictures from the GOP structure.

2. Recombination Method 2

In the second method for recombining the video stream, the transport stream is modified using a PID filter. The PID filter can be implemented as part of the demodulator, as shown in FIG. 5, or as part of the demultiplexer.

For illustrative purposes, in a multi-program transport stream, each program can include an I-PID for the I-picture, the PRED-PID for the predicted pictures, an audio PID, and a data PID. Any packet with a PID that matches any of the PIDs in the desired program, as identified by the program mapping table (PMT) has its PID modified to the lowest PID in the program (the PID that is referenced first in the program's PMT). As a specific example, a program can include an I-PID of 50 and a PRED-PID of 51. For this program, the PID-filter modifies the PRED-PID to 50 and thereby, the I and predicted access units attain the same PID number and become a portion of a common stream. As a result, the transport stream from the PID filter contains a program with a single video stream having packets that appear in the proper order to be decoded as valid MPEG bitstream.

Note that the incoming bit stream does not necessarily contain any packets with a PID equal to the lowest PID referenced in the program's PMT. Also note that it is possible to modify the PIDs to other PID numbers than lowest PID without changing the operation of the process.

When the PIDs of incoming packets are modified to match the PIDs of other packets in the transport stream, the continuity counters of the merged PIDs may become invalid at the merge points, since each PID has its own continuity counter. For this reason, the discontinuity indicator in the adaptation field is set for any packets that may immediately follow a merge point. Any decoder components that check the continuity counter for continuity properly processes the discontinuity indicator bit.

FIG. 7 is a flow diagram of an embodiment of a second recombination process 700. At step 710, the process waits for a (viewer) selection an I-PID to be received. The I-PID, comprising the first picture of a video stream's GOP, identifies the stream to be received. A packet having the selected I-PID is then detected.

At step 715, the PID of the I stream is re-mapped to a particular number (e.g., PID*). At this step, the PID filter modifies all PIDs of the desired I-stream packets to PID*. At step 720, the PID number of the predicted pictures (PRED-PID) is also re-

mapped to PID* by the PID filter, which modifies all PIDs of the PRED-PID packets to PID*.

At step 725, the packets of the PID* stream are extracted from the transport stream by the demultiplexer. At step 730, the payloads of the packets that include the video stream header information and the I and predicted picture data are coupled to the video decoder as video information stream V. It should be noted that the packets are ordered in the transport stream in the same order as they are to be decoded.

At step 735, a query is made whether a different picture (e.g., another IPG page) is requested. If a different picture is not requested, then the process returns to step 710 where the demultiplexer waits for the next packets having the identified I-PID. Otherwise, if a different picture is requested, then the I-PID of the new desired picture is identified at step 740 and the process returns to step 710.

The process shown in FIG. 7 is used to produce an MPEG-compliant video stream by merging the I stream and predicted stream before the demultiplexing process.

3. Recombination Method 3

The third recombination method accomplishes MPEG bitstream recombination by using splicing information in the adaptation field of the transport packet headers and by switching between video PIDs based on splice countdown concept.

In the third recombination method, the MPEG streams signal the PID-to-PID switch points using the splice countdown field in the transport packet header's adaptation field. When the PID filter is programmed to receive one of the PIDs in a program's PMT, the reception of a packet containing a splice countdown value of 0 in its header's adaptation field causes immediate reprogramming of the PID filter to receive another video PID. It should be noted that special attention to splicing syntax is required for systems that use splicing for other purposes.

FIG. 8 is a flow diagram of an embodiment of a third recombination process 800. At step 810, the process waits for a (viewer) selection of the I-PID to be received for the desired IPG page. The I-PID, comprising the first picture of a stream's GOP, identifies the stream to be received. A packet having the selected I-PID is then detected .

At step 815, the I-PID packets are extracted from the transport stream until, and including, the I-PID packet with a slice countdown value of zero. At step 820,

the payloads of the packets that include the header information related to the video stream and the intra-coded slices are coupled to the video decoder as video information stream V.

At step 825, the PID filter is re-programmed to receive the predicted picture (PRED-PID) packets. At step 830, the predicted picture packets (e.g., PID1 in FIG. 2) are extracted from the transport stream. At step 835, the payloads of the packets that include the header information related to the video stream and the predictive-coded pictures are coupled to the video decoder. At the end of step 835, a complete GOP, including the I-picture and the predicted picture data are coupled to the video decoder as video stream V. As the payloads are sent to the video decoder in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination processing.

At step 840, a query is made whether a different picture (e.g., another IPG page) is requested. If a different picture is not requested, the process proceeds to step 850 where the PID filter is re-programmed to receive the previous desired I-PID. Otherwise, if a different picture is requested, then the I-PID of the new desired picture is identified at step 845 and the process proceeds to step 850 where the PID filter is re-programmed to receive the new I-PID. The process then returns to step 810, where the demultiplexer waits for the next packets having the PID of the desired picture.

The process shown in FIG. 8 can be used to produce an MPEG-compliant video stream, where the PID-to-PID switch is performed based on a splice countdown concept.

C. INTERACTIVE PROGRAM GUIDE

FIG. 9 depicts an example of an IPG page 900 in accordance with an embodiment of the invention. In the specific embodiment shown in FIG. 9, IPG page 900 includes a time slot region 905, a guide region 910, a video region 920, an icon region 940, a program description region 950, a logo region 960, and a date/time region 970. Other designs for the IPG page with different layouts, configurations, and combinations of regions and objects can be contemplated and are within the scope of the invention.

Time slot region 905 includes a first time slot object 905a and a second time slot object 905b that indicate the time slots for which program guide is being provided on the IPG page. Guide region 910 is used to display program listing for a

group of channels. In the embodiment shown in FIG. 9, the program listing shows the available programming in two half-hour time slots. Guide region 910 thus includes a number of channel objects 912a through 912j used to display channel information for the listing of channels. Guide region 910 further includes a pair of channel indicators 914a and 914b that identifies the current cursor location.

Program description region 950 is used to present descriptive information relating to a particular program selected from the program listing, or may be used to show other information. Video region 920 may be used to display images, videos, text, or a combination thereof, which may be used for advertisements, previews, or other purposes. Video region 920 may be implemented as described above in a server-centric manner. Logo region 960 may include a logo of a service operator or other entity and may be optionally displayed. Date/time region 970 may be configurable by the user and may also be optionally displayed.

Icon region 940 is used to display various icons, which may be created and/or enabled by the user. Each icon in icon region 940 can represent a filter or a link to another IPG page or a particular interface. Each filter selects a particular type of programming to be included in the program listing shown in guide region 902. For example, a Pay Per View (PPV) icon 941 may be a filter that selects only PPV programming to be included in the program listing. A Favorites icon 942 may be a filter that selects only channels designated by the user to be among his or her favorites. A Movies icon 943 may be a filter that selects only movies or movie channels. A Kids icon 944 may be a filter that selects only channels for children or programming appropriate or produced for viewing by children. A Sports icon 945 may be a filter that selects only sports channels or sports-related programming. A Music icon 946 is a link to a music interface. An Options icon 947 may also be a link to a menu of IPG options that the user may select amongst. Such options may include (1) configuration and selection/deselection information of IPG related services, (2) custom information such as deactivating some of the filters or accessing the custom condensed listing menus, and other features and functionality. A Weather icon 948 may be a link to an interface to weather information.

In a system, illustratively, comprising 100 channels of information, the channels can be displayed in 10-channel groups having associated with them two half-hour time slots. In this organization, ten or more video PIDs can be provided to send the

present-time channel/time/title information, one or more audio PIDs can be provided to send the audio barker, and/or one or more data PIDs (or other data transport method) can be provided to send the program description data, overlay data, and the like. To fully broadcast interactive program information for up to 24 hours in advance, 240 (e.g., 10•24) or more video PIDs can be provided, along with one or more audio PIDs and, optionally, one or more data PIDs.

The time depth of a program guide is defined by the amount of time programming is provided for in the broadcast video PIDs for a particular channel group. The channel depth of the program guide is defined by the number of channels available through the guide (as compared to the total number of channels in the system). In a system providing only half of the available channels via the broadcast video PIDs, the channel depth 50%. In a system providing 12 hours of "look-ahead" time slots, the time depth is 12 hours. In a system providing 16 hours of "look-ahead" time slots and 4 hours of "look-back" time slots, the time depth is +16/-4 hours.

The video streams representing the IPG are sent in one or more transport streams, within the form of a single or multi-program as described below. A user desiring to view the next 1-hour time interval (e.g., 10:00 – 11:00) may activate a "scroll right" object (or move the joystick to the right when a program within guide region 910 occupies the final displayed time interval). Such activation results in a controller within the terminal noting that a new time interval is desired. The video stream for the new time interval is then decoded and displayed. If the desired video stream is within the same transport stream (i.e., another PID), then the video stream is simply decoded and presented. If the desired video stream is within a different transport stream, then that transport stream is extracted from the broadcast stream and the desired video stream is decoded and presented. And if the desired transport stream is within a different broadcast stream, then that broadcast stream is tuned, the desired transport stream is extracted, and the desired video stream is decoded and presented.

A viewer interaction requesting a prior time interval or a different set of channels results in the retrieval and presentation of the desired video stream. If the desired video stream is not part of the broadcast video streams, then a pointcast or demand-cast session, for example, may be initiated as described in U.S. Patent Application Serial No. 09/539,228, entitled "MESSAGING PROTOCOL FOR DEMAND-CAST SYSTEM AND BANDWIDTH MANAGEMENT," filed March 30,

2000, assigned to the assignee of the invention and incorporated herein by reference. For this pointcast session, the terminal sends a message to the head-end via a back channel requesting a particular stream. The head-end processes the request, retrieves the desired stream from an information server, and incorporates the stream within a transport stream as another video PID. Preferably, the desired stream is inserted into the transport stream currently being tuned/selected by the terminal. The head-end further informs the terminal which PID should be received and from which transport stream it should be demultiplexed. The terminal then retrieves the desired video PID. If the video PID is within a different transport stream, the terminal first demultiplexes that transport stream (possibly by tuning a different broadcast stream within the forward channel).

Upon completion of the viewing of the desired stream, the terminal can indicate to the head-end that it no longer needs the stream. In response, the head-end can tear down the pointcast or demand-cast session. The terminal can then return to the broadcast stream from which the pointcast session was launched.

SLICE-LEVEL PROCESSING

D. ENCODING

Various data structures can be used to represent data for the IPG and various encoding schemes can be used to encode the IPG pages such as the one shown in FIG. 9. For an interactive information distribution system, program guide data may be processed and sent over a number of elementary streams. Each elementary stream carries a video sequence comprised of a sequence of pictures. Each picture can include a combination of textual and video information (e.g., text on the left side of the picture and video on the right side). Depending on the particular implementation and operation of the interactive information distribution system, some of the pictures may include common (i.e., redundant) information. The invention provides a number of efficient data structures for use in a number of IPG applications to reduce the amount of data needed to represent a group of video sequences having some common textual and/or video information.

1. Data Structures

FIG. 10A depicts a matrix representation 1000 of program guide data for a group of IPG pages. In this representation, the horizontal axis represents the video

sequences to be transmitted, and the vertical axis represents time indices for the video sequences. In this specific example, ten video sequences are generated and labeled as IPG pages 1 through 10. Each video sequence is composed of a time sequence of pictures. In this specific example, 15 time indices are shown on the vertical axis and labeled as t_1 through t_{15}. Each group of 15 pictures for each video sequence forms a group of pictures (GOP) for that video sequence.

As shown in FIG. 10A, the program guide data is represented using a matrix 1000 that is a two-dimensional array of elements. In the embodiment shown in FIG. 10A, each element of matrix 1000 includes two regions (or portions) - a guide portion and a video portion. For example, the element in the first column of the first row represents the guide portion (g_1) and video portion (v_1) of IPG page 1 at time index t_1, the element in the second column of the first row represents the guide portion (g_2) and video portion (v_1) of IPG page 2 at time index t_1, and so on.

Matrix 1000 in FIG. 10A is illustratively shown to include ten GOPs for ten IPG pages. However, matrix 1000 can be designed to have any defined dimension (i.e., an MxN dimension, where M is the number of IPG pages or video sequences and N is the number of pictures in the GOP, and M and N can each be any integer one or greater).

In the specific example shown FIG. 10A, the guide portion for each IPG page is different but the video portion is common for all ten IPG pages. Thus, the guide portion index (g_1, g_2, ..., g_{10}) increases in number, corresponding to the IPG pages, as the matrix is traversed across the horizontal axis. Because the video portion is common for all IPG pages, the video portion index (e.g., v_1) remains constant as the matrix is traversed in the horizontal axis. In this example, the guide portion is static over the GOP but the video portion changes over time (e.g., for a moving video). Thus, the guide portion index remains constant as the matrix is traversed in the vertical time axis, but the video portion index changes with the time index.

As noted above, each of the ten video sequences in FIG. 10A includes 15 pictures that can be coded as a group of pictures. For example, the video sequence for IPG page 1 can be encoded as a GOP comprised of the 15 coded pictures: I1, B1, B1, P1, B1, B1, P1, B1, B1, P1, B1, B1, P1, B1, and B1, where I represents an intra-coded picture, P represents a un-directionally predictive-coded picture, and B represents a bi-directionally predictive coded picture.

FIG. 10B depicts an embodiment of a data structure 1030 that can be used to reduce the amount of data to be coded and delivered to the terminals. Data structure 1030 includes a group of intra-coded pictures 1032 and a group of predictive-coded pictures 1034 that can be used to fully represent the data in data structure 1030. In an embodiment, intra-coded picture group 1032 includes ten intra-coded pictures at time index t_1 for the ten IPG pages. These intra-coded pictures can be assigned to I-PIDs 1 through 10. The I-PID for IPG page 1 includes the guide portion (g_1) and the video portion (v_1), the I-PID for IPG page 2 includes the guide portion (g_2) and the video portion (v_1), and so on. In an embodiment, predictive-coded picture group 1034 includes 14 predictive-coded pictures of one of the IPG pages for time indices t_2 through t_{15}. The predictive-coded picture group 1034 is also assigned a PID (e.g., base-PID or PRED-PID). For example, if IPG page 1 is the selected picture as shown in FIG. 10B, the base-PID may comprise the following picture sequence: B1, B1, P1, B1, B1, P1, B1, B1, P1, B1, B1, P1, B1, and B1.

Using data structure 1030 shown in FIG. 10B, instead of processing all 150 pictures for matrix 1000, the number of pictures to be coded and delivered reduces to 24. This reduction in transmitted data is achieved without loss of information. The reduction in the required bit rate can be computed for a specific example in which 40 percent of a GOP's bits is assigned to an I-picture (e.g., the I-PID) and the remaining 60 percent is assigned to the 14 remaining P and B-pictures (e.g., the base-PID). Data structure 1030 can then reduce the relative bit rate from 1500 (i.e., 10 I-pictures x 40 + 10 base-PID x 60 = 1000) down to 460 (i.e., 10 I-pictures x 40 + 1 base-PID x 60 = 460). The reduction in bit rate can then be used to transmit more video sequences (e.g., more IPG pages) with the same common video portion.

If a viewer wants to view the guide data for a particular group of channels, a demultiplexer at the terminal selects the related I-PID and recombines the selected I-PID with the base-PID to produce a recombined stream, which is then decoded by the video decoder.

FIG. 10C depicts an embodiment of a data structure 1060 that can be used to further reduce the amount of data to be coded and delivered to the terminals. In the illustrated example, ten IPG pages are available, with each page represented by a guide portion (g) and a common video portion (v). For example, IPG page 1 is represented by g_1/v_1, IPG page 2 is represented by g_2/v_1, and so on. In data structure 1060, ten guide

portions g_1 through g_{10} are associated with the first video portion (v_1). Each portion can be slice-based encoded as described below.

FIG. 10C also illustrates an exemplary assignment of PIDs to various portions of the IPG pages. In FIG. 10C, only the content that is assigned a PID is delivered to the terminals. The intra-coded guide portions g_1 through g_{10} are assigned to PID1 through PID10, respectively. One of the common intra-coded video portion v_1 (e.g., IPG page 10) is assigned to PID11. In this form, substantial bandwidth saving is achieved by delivering the intra-coded video portion v_1 only once. Finally, the predictive-coded pictures g_1/v_2 through g_1/v_{15} are assigned to PID12. As shown in FIG. 10C, a substantial saving in bandwidth is achieved by transmitting only one group of fourteen predictive-coded pictures, g_1/v_2 through g_1/v_{15}. The PID assignment and decoding processes are described in further detail below.

The matrix representations described in FIGS. 10A through 10C may be used to represent program guide data with different contexts such broadcast, narrowcast, pointcast, shared pointcast, and others.

E. SLICE-LEVEL PROCESSING

1. <u>Encoding Slices</u>

To enhance error recovery, the MPEG-2 standard contemplates the use of a "slice layer" in which a video picture is divided into one or more slices. A slice contains a sequence of one or more contiguous macroblocks. The sequence can begin and end at any macroblock boundary within a picture. An MPEG-2 decoder, when provided a corrupted bitstream, uses the slice layer to avoid reproducing a completely corrupted picture. For example, if a corrupted bitstream is decoded and the decoder determines that the present slice is corrupted, the decoder skips to the next slice and begins decoding. As such, only a portion of the reproduced picture is corrupted.

In accordance with the MPEG-2 standard, each slice includes one or more macroblocks. (A picture may consist of 27 rows and 22 columns of macroblocks.) Each macroblock is defined as a rectangular group of picture elements (pixels). A slice may start at any macroblock location in a picture and extend from left-to-right and top-to-bottom through the picture. The stop point of a slice can be chosen such that any macroblock can be the start or end boundary. The slice layer syntax and its use in

forming an MPEG-2 bitstream is known to those skilled in the art and not described herein.

In accordance with an aspect of the invention, the IPG pages can be encoded at the slice layer to achieve greater flexibility in the encoding process and improved compression efficiency. A slice-based encoding system enables the guide and video of the IPG to be efficiently coded and flexibly transmitted, as described below. Consequently, a viewer can easily and quickly move from one IPG page to another.

The slice-based encoding technique separately encodes the guide and video portions of the IPG page. As such, the guide and video portions can each be represented by one or more different slices.

FIG. 11 illustrates an exemplary slice division of IPG page 900 shown in FIG. 9 in which the guide portion and the video portion are each divided into N slices (e.g., g/s_1 through g/s_N for the guide portion, and v/s_1 through v/s_N for the video portion). Each slice includes a number of macroblocks. For example, if there are 22 macroblocks per row for the IPG page, then each portion may include 11 macroblocks per row.

The slices in the guide portion can be pre-encoded to form a "slice form grid page" database that contains a number of encoded slices of the guide portion. In this implementation, the guide slices can be recalled from the database and flexibly combined with the separately encoded video slices to form the IPG page. Alternatively, the encoding process for the guide portion can also be performed real-time during the broadcast process. The IPG is transmitted to the line neighborhood equipment and, ultimately, to the terminals. The line neighborhood equipment may be designed and operated to assemble the IPG data for the neighborhood, as described below.

Although the following description of the slice-based encoding technique is presented in the context of IPG, slice-based encoding is equally applicable to a broad range of applications, such as broadcast video-on-demand, e-commerce, Internet, video education services, and others. Slice-based encoding is especially advantageous for delivery of video sequences with common content.

FIG. 13 depicts a process 1300 that can be used to form a bitstream 1310 that includes the intra-coded slices encoded at time index t_1 in FIG. 10C. At step 1302, a number of IPG pages 1302a through 1302j are provided to the encoding unit. At step 1304, each IPG page is slice-based encoded to form, for example, the guide portion slices g_1/s_1 through g_1/s_N and the video portion slices v/s_1 through v/s_N for the IPG page.

The slice-based encoding process for the guide and video portions can be performed based on various encoding schemes. For example, the guide slices can be pre-encoded by a software MPEG-2 encoder or encoded by the same encoder used to encode the video portion. If the same encoder is employed, the parameters of the encoding process can be adjusted dynamically for the two portions. Regardless of the encoder implementation and encoding parameters, each portion is encoded independently. In encoding the video portion, the encoding can be performed assuming a full picture size (i.e., a picture covering both the guide and video portions) with the guide portion of the full picture being padded with null data. Step 1304 is performed at the head-end.

At step 1306, the encoded guide and video portion slices are sent to the line neighborhood equipment. If the line neighborhood equipment is implemented as part of the head-end, then the encoded slices are delivered to the line neighborhood equipment in a packetized elementary stream (PES) format or a similar format as the output of the video encoders. If the line neighborhood equipment is implemented as a remote network equipment, the encoded slices are formatted into a form suitable for delivery over a network (e.g., via a cable modem protocol or some other method). Once the slice-based streams are available at the line neighborhood equipment, the slice combiner at step 1306 orders the slices into a form suitable for decoding at the terminals.

As depicted in part (b) of FIG. 13, the guide and video slices are ordered in a manner as if the original pictures in part (a) of FIG. 13 were scanned in a left-to-right and top-to-bottom order. Each of the slice packets is then assigned to an appropriate PID by the multiplexer, as described in below. For example, PID1 can be assigned to guide slices g_1/s_1 through g_1/s_N, PID2 can be assigned to guide slices g_2/s_1 through g_2/s_N, and so on, PID10 can be assigned to guide slices $g_{10}/s1$ through g_{10}/s_N, and PID11 can be assigned to video slices v/s_1 through v/s_N. The resultant transport stream containing the intra-coded guide and video slices is illustrated in part (c) of FIG. 13. Based on this transport stream structure, a receiving terminal retrieves the original picture by reconstructing a video picture row-by-row. For example, if PID1 is desired, the terminal first retrieves the guide slice g_1/s_1 assigned PID1 then the video slice v/s_1 assigned PID11, next retrieves the guide slice g_1/s_2 assigned PID1 then the video slice v/s_2 assigned PID11, and so on.

FIG. 14 illustrates a process 1400 for producing a bitstream 1408 that includes the slices for the predictive-coded pictures accompanying the transport stream

generation process 1300 described in FIG. 13 for the intra-coded pictures. As shown in FIG. 10C, illustratively, only the predictive-coded slices belonging to IPG page 1 are delivered.

At step 1402, the predictive-coded slices are generated at the head-end independently and then forwarded to a line neighborhood equipment located locally or in a remote network location. At step 1404, slices in the predictive-coded guide and video portions (e.g., from time periods t_2 through t_{15}) are scanned from left-to-right and top-to-bottom in slice-combiner and the complete data are assigned PID12 by the multiplexer. It can be noted that the guide slices g_1/s_1 through g_1/s_N at each time period t_2 through t_{15} do not change from their corresponding intra-coded slices at time period t_1. Therefore, these slices can be coded as skipped macroblocks "s_K". Conventional encoding systems do not necessarily skip macroblocks in a region even when there is no change from picture to picture. In order to provide this functionality, the encoder is given the parameters for the slices to skip macroblocks without any further encoding evaluations. At step 1406, the slice packets are ordered into a portion of a final transport stream. In an embodiment, the final transport stream first includes the video slice packets for time periods t_2 through t_{15} (i.e., v_2/s_1 through v_2/s_N for t_2, and so on, and v_{15}/s_1 through v_{15}/s_N for t_{15}), then includes the skipped guide slices s_K/s_1 through s_K/s_N from time periods t_2 through t_{15}.

FIG. 15 illustrates a process 1500 for producing a predictive-coded slice bitstream 1506 in accordance with another embodiment of the invention. Process 1500 is an alternative embodiment to process 1400 in FIG. 14, which scans the skipped guide portion and video portion separately. At step 1502, the predictive-coded slices are produced. At step 1504, the coded slices are scanned to intersperse the "skipped" slices (s_K) with the video slices (v/s). In process 1500, the slices are scanned from left-to-right and top-to-bottom completely, including the skipped guide and video data. As such, at step 1508, bitstream 1506 has the skipped guide and video slices distributed uniformly throughout the transport stream.

FIG. 16 depicts an MPEG-compliant transport stream 1600 that includes the complete information needed by a decoder at the terminal to recreate IPG pages that were slice-based encoded. Transport stream 1600 comprises intra-coded bitstream 1310 for the guide and video slices (PID1 to PID11), a number of audio packets 1602 identified by an audio PID, and bitstream 1508 containing the predictive-coded slices in PID12. The rate of audio packet insertion between video packets is determined based on

the audio and video sampling ratios. For example, if audio is digitally sampled at one tenth of video sample rate, then an audio packet may be inserted into the transport stream for every ten video packets. Transport stream 1600 may also contain, illustratively after every 64 packets, data packets that carry overlay updates, raw data, HTML, java, URL, instructions to load other applications, user interaction routines, and the like, to the terminals. Data PIDs are assigned to different set of data packets related to the guide slice sets and also the video slice sets.

The above encoding embodiments assumed that the IPG page was divided into one guide portion and one video portion. For example, in FIG. 11, the guide portion is defined as the left half of the IPG page and the video portion is defined as the right half of the IPG page. However, the invention can be extended to have one or more guide portions and one or more video portions. Each video portion may contain video having different rates of motion or a stationary image. For example, the first portion may have a rate of 27 frames per second, and the second and third portions may each have a rate of 2 frames per second.

FIG. 17A illustrates an embodiment of an IPG page 1700 having a guide portion 1702 and three video portions 1704, 1706 and 1708. To encode IPG page 1700, each portion is separately encoded and assigned a respective PID.

FIG. 17B illustrates an assignment map for encoding each portion of IPG page 1700 shown in FIG. 17A. Guide portion 1702 is encoded as slices g/s_1 through g/s_N, the first video portion 1704 is encoded as slices v_A/s_1 through v_A/s_M, the second video portion 1706 is encoded as slices v_B/s_{M+1} through v_B/s_L, and the third video portion 1708 is encoded as slices v_C/s_{L+1} through v_C/s_N.

FIG. 18 depicts a scanning process 1800 used to produce a bitstream 1810 that includes the intra-coded slices for IPG page 1700 shown in FIG. 17B. Scanning process 1800 scans from left-to-right and from top-to-bottom through the slices shown in FIG. 17B. As the encoded IPG is scanned, PIDs are assigned to the slices. In this example, the guide portion slices for the 10 IPG pages in time period t_1 (see FIG. 10C) are assigned PID1 through PID10. The first video portion slices are assigned PID11, the second video portion slices are assigned PID12 and the third video portion slices are assigned PID13.

At step 1802, slices 1 through M are processed, and the guide slices are assigned PID1 through PID10 and the first video portion slices are assigned PID 11. At

step 1804, slices M+1 to L are processed, and the second video portion slices are assigned PID 12. And at step 1806, slices L+1 to N are processed, and the third video portion slices are assigned PID 13. The resultant bitstream 1810 contains the PIDs for slices 1 through M, followed by the PIDs for slices M+1 through L, and lastly by the PIDs for slices L+1 through N.

FIG. 19 depicts a process 1900 for assigning PIDs to the predictive-coded slices for IPG page 1700 shown in FIG. 17B. The scanning process is performed, at step 1902, from left-to-right and from top-to-bottom through the v_A, v_B, and v_C predictive-coded slices. PIDs are assigned such that the v_A video slices are assigned PID 11, the v_B video slices are assigned PID12, and the v_C slices are assigned PID13.

After the video predictive-coded slices have been assigned PIDs, the skipped slices are also assigns PIDs, at step 1904. The skipped guide slices that vertically correspond to the v_A video slices are assigned PID14, the skipped guide slices that vertically correspond to the v_B video slices are assigned PID15, and the skipped guide slices that vertically correspond to the v_C video slices are assigned PID16. At step 1908, the resultant predictive-coded bitstream 1910 comprises the predictive-coded video slices 1912 and the skipped slices 1914. Bitstream 1810 of intra-coded slices (FIG. 18) and bitstream 1910 of predictive-coded slices (FIG. 19) are combined into a transport stream having a form similar to that shown in FIG. 16.

To change pages in the guide, it is desirable to be able to switch between programs (e.g., video PIDs for groups of slices) in a seamless manner. This is not easily achievable using a standard channel change with the terminal switching directly from PID-to-PID, because such operation normally flushes the video and audio buffers and typically result a blank screen for half a second.

To provide seamless switching at the decoder, a splice countdown (or random access indicator) method is employed at the end of each video sequence to indicate the point at which the video should be switched from one PID to another.

Using the same profile and a constant bit rate for coding, the video and guide encoding units generate streams for different IPG pages having similar lengths compared to each other. This is due to the fact that the source material is almost identical, and differs only in the characters in the guide from one IPG page to another. Thus, while the streams are generated having approximately equal lengths, they typically do not have exactly equal lengths. For example, for any given sequence of 15 video

pictures, the number of transport packets in the sequence typically varies from one IPG page to another. Thus, a fine adjustment is used to synchronize the beginnings and ends of the sequences across all IPG pages to support the operation of the splice countdown switching method.

An aspect of the invention provides techniques to synchronize a number of streams to enable seamless switching at the terminal. Three synchronization methods are provided.

In the first synchronization method, for each (e.g., 15-picture) sequence, the multiplexer in the line neighborhood equipment identifies the length of the longest IPG page for that particular sequence. The line neighborhood equipment then adds sufficient null packets to the end of each IPG page so that all IPG pages have the same length. The multiplexer then adds switching packets at the end of the sequence, after the null packets.

The second synchronization method uses buffering for all packets for all IPG pages for each (e.g., 15-picture) sequence. The buffered packets can be ordered in the transport stream such that the packets for each IPG page can appear at slightly higher or lower frequencies, so that the IPG pages all finish at the same point. Switching packets are then added by the multiplexer in the line neighborhood equipment at the end of each stream, which does not include the null padding.

The third synchronization method starts each sequence together and then waits until all packets for all IPG pages have been generated. Once the generation of all packets is completed, switching packets are placed in the streams at the same time and point in each stream.

Depending on the implementation of the decoder within the terminal and the requirements of the application being supported, each of the above synchronization methods can be advantageously applied. For example, the first synchronization method, which uses null padding, can be applied to avoid bursts of N packets of the same PID into a decoder's video buffer faster than the MPEG specified rate (e.g., 1.5 Mbit).

The above synchronization methods can be applied to other synchronization applications, and can be used to derive other methods for synchronizing the streams for seamless switching.

F. MULTIPLEXING STRUCTURES, LATENCY REDUCTION, and STREAM INDEXING

1. Level Zero, Level One, and Level Two Encoding

As shown in FIG. 10A, in the basic ensemble data structure 1000, each of the video sequences is encoded independently in a vertical dimension and assigned a separate PID. In this encoding structure, the ten coded video streams assigned PIDs 1-10 contain redundant information that is included in the delivered transport stream. In particular, ten video pictures (with each video picture including the guide and video portions) are sent in parallel for each time period. In the description below, this first encoding technique is referred to as "level zero" encoding.

As shown in FIG. 10B, in data structure 1030, a substantial portion of the redundancy is removed. Using only elements 1032a through 1032j and 1034, all elements in each row and column of the matrix may be reconstructed. While ten video pictures (with each video picture including the guide and video portions) are sent for the intra-coded time period t_1, only one video picture (including the guide and video portions) is sent for the predictive-coded time periods t_2 through t_{15}. In the description below, this second encoding technique is referred to as "level one" encoding.

As shown in FIG. 10C, in encoding structure 1060, redundancy is further removed by dividing each picture into portions, encoding each portion as slices, and transmitting the unique slices. These slices are later appropriately recombined to regenerate the pictures. In the description below, this third encoding technique is referred to as "level two" encoding.

In each of these three encoding techniques, the elementary streams are multiplexed as described below.

2. Multiplexing Structures, Program Mapping, and Transport Stream Formation

FIG. 20 is a block diagram illustrating an apparatus for encoding, packetizing, multiplexing, and assigning programs to video, audio, and data in accordance with a "level zero" embodiment of the invention. As described above, the "level zero" embodiment delivers ten video pictures for each time period (in addition to an audio

signal). Apparatus 2000 includes an encoding and packetizing unit 2002 and a transport stream multiplexer and program map table (PMT) assigner 2004.

In the example shown in FIG. 20, for each time period, encoding and packetizing unit 2002 receives ten video sequence inputs 2006, one audio input 2008, and ten data inputs 2010. Encoding and packetizing unit 2002 encodes and packetizes each of these inputs. In this example, encoding and packetizing unit 2002 outputs ten video streams 2012, one audio stream 2014, and ten data streams 2016.

In this example, each video input is encoded independently and packetized into a respective video stream. The ten video inputs 2006 are encoded by aligning the pictures of the video inputs to each other so that each group of pictures (GOP) starts at approximately the same time point for each input. Each output video stream 2012 is assigned a respective video PID. The single common audio input is also encoded and packetized into a separate audio stream, which is assigned an audio PID. In addition, the ten data inputs are packetized into ten separate data streams, with each data stream being assigned a respective data PID.

Transport stream multiplexer and PMT assigner 2004 receives the outputs from encoding and packetizing unit 2002. In this example, transport stream multiplexer and PMT assigner 2004 receives the ten video streams 2012, one audio stream 2014, and ten data streams 2016. Transport stream multiplexer and PMT assigner 2004 multiplexes the received streams to form one or more final transport streams 2018. In the case of a single final transport stream, one packet of each (video, audio, and data) stream may be sequentially time multiplexed to form the final transport stream. For example, a packet from video stream 1, then a packet from video stream 2, then a packet from video stream 3, and so on, can be multiplexed into the final transport stream.

Transport stream multiplexer and PMT assigner 2004 also provides packets conveying a program mapping table (PMT). The PMT specifies packet identifier (PID) values for program components. For example, a program may correspond to a particular broadcast channel, and the PMT may specify the PID values for the video, audio, and data relating to that broadcast channel. The packets conveying the PMT are also included in final transport stream(s) 2018.

FIG. 21A is a diagram illustrating a program assignment structure 2100 for a single final transport stream with multiple programs in accordance with an embodiment

of the invention. Program assignment structure 2100 assigns to each program a video PID, an audio PID, and a data PID.

In this example, for each program, the video PID is one of ten video PIDs, the audio PID is the same for each program, and the data PID is one of ten data PIDs. For example, program 1 2101 is assigned video PID1, the audio PID, and data PID1, program 2 2102 is assigned video PID2, the audio PID, and data PID2, and so on, and program 10 2110 is assigned video PID10, the audio PID, and data PID10. It can be noted that although the audio PID is referenced for every program, the audio packets are multiplexed into final transport stream 2018 only once.

FIG. 21B is a diagram illustrating a program assignment structure 2150 for a final transport stream with a single program in accordance with a "level zero" embodiment of the invention. In this example, program assignment 2150 assigns to single program 2152 the ten video PIDs, the audio PID, and the ten data PIDs. This assignment results in a reduced number of programs.

FIG. 22 is a diagram illustrating the multiplexing of video, audio, and data packets into a final transport stream in accordance with a "level zero" embodiment of the invention. In this example, video packets 2202 include packets with video PIDs 1-10, audio packets 2204 include packets with the audio PID, and data packets 2206 include packets with data PIDs 1-10.

Transport stream multiplexer 2004 multiplexes these various packets into one or more final transport streams 2200. In the example shown in FIG. 22, the packets are multiplexed into a single final transport stream 2200. As shown, for example, the video and audio packets may be interleaved and the data packets may be arranged separately from them.

In particular, since audio typically has a lower rate compared with video (e.g., one tenth the video rate), the audio packets may be inserted into final transport stream 2200 illustratively every 10th video packet. Similarly, data typically also has a lower rate compared with video. Hence, for example, 64 video/audio packet groups 2208 may be sent sequentially, followed by a single data packet group 2210, followed by another 64 video/audio packet groups 2208, followed by another data packet group 2210, and so on. The number of video/audio packet groups sent sequentially may be adjusted depending on the data rate in comparison to the video/audio rate.

FIG. 23 is a diagram illustrating an assignment structure 2300 for multiple final transport streams in accordance with a "level zero" embodiment of the invention. In this example, assignment structure 2300 assigns the various video, audio, and data packets to three transport streams. Also, in this specific example, transport stream 1 2302 is assigned video PIDs 1-3, the audio PID, and data PIDs 1-3. Transport stream 2 2304 is assigned video PIDs 4-6, the audio PID, and data PIDs 4-6. And transport stream 3 2306 is assigned video PIDs 7-10, the audio PID, and data PIDs 7-10. The particular assignment structure selected depends on the number of PIDs and the number of transport streams. Unlike this example, in a preferred embodiment, the number of video PIDs is evenly divisible by the number of transport streams.

In addition, different program assignments may be imposed on each final transport stream to yield a single program or multiple programs in a manner analogous to that described above for FIGS. 21A and 21B.

FIG. 24 is a diagram illustrating a final transport stream 2400 in accordance with a "level one" embodiment of the invention. As described above, the "level one" embodiment sends ten video pictures for each intra-coded time period (t_1), but only one video picture for each predictive-coded time period. Final transport stream 2400 in FIG. 24 includes intra-coded packets 2402 and predictive-coded packets 2404.

Intra-coded packets 2402 may include, for example, 64 sequential video/audio packet groups, followed by a data packet group, much like final transport stream 2200 shown in FIG. 22. These intra-coded packets 2402 include information from intra-coded pictures 1032a through 1032j in FIG. 10B. However, unlike final transport stream 2200 shown in FIG. 22, final transport stream 2400 of FIG. 24 only includes packets for intra-coded pictures. For predictive-coded pictures, final transport stream 2400 includes predictive-coded packets 2404, which carry information relating to predictive-coded pictures 1034 in FIG. 10B.

In addition, different program assignments may be imposed on the final transport stream to yield a single program or multiple programs in a manner analogous to that described above for FIGS. 21A and 21B.

FIGS. 25A and 25B are diagrams illustrating multiple final transport streams in accordance with a "level one" embodiment of the invention. The example illustrated in FIGS. 25A and 25B includes three final transport streams: a first final transport stream 2502, a second final transport stream 2504, and a third final transport

stream 2506. Each final transport stream includes intra-coded packets and predictive-coded packets.

Intra-coded packets 2512 for first final transport stream 2502 include video/audio packet groups 2532. Each video/audio packet groups 2532 includes, in this example, ten video packets with video PIDs 1-3 and an audio packet with the audio PID. For example, 64 video/audio packet groups 2532 may be serially included in first final transport stream 2502, followed by a group of data packets with data PIDs 1-3, and followed by predictive-coded packets 2522.

Similarly, intra-coded packets 2524 for second final transport stream 2504 include video/audio packet groups 2534. Each video/audio packet groups 2534 includes, in this example, ten video packets with video PIDs 4-6 and an audio packet with the audio PID. For example, 64 video/audio packet groups 2534 may be serially included in second final transport stream 2504, followed by a group of data packets with data PIDs 4-6, and followed by predictive-coded packets 2524.

Finally, intra-coded packets 2526 for third final transport stream 2506 include video/audio packet groups 2536. Each video/audio packet groups 2536 includes, in this example, ten video packets with video PIDs 7-10 and an audio packet with the audio PID. For example, 64 video/audio packet groups 2536 may be serially included in third final transport stream 2506, followed by a group of data packets with data PIDs 7-10, and followed by predictive-coded packets 2526.

Again, the particular assignment structure selected for use may depend on the number of PIDs and the number of transport streams. In addition, different program assignments may be imposed on each final transport stream to yield a single program or multiple programs in a manner analogous to that described above for FIGS. 21A and 21B.

FIG. 26 is a diagram illustrating a final transport stream 2600 in accordance with a "level two" embodiment of the invention. As described above, the "level two" embodiment divides each picture into slices and transmits the unique slices. The received slices are later appropriately recombined to regenerate the pictures. Final transport stream 2600 in FIG. 26 includes guide slice packets 2602, intra-coded video slice packets 2604, audio packets 2606, data packets 2608, and predictive slice packets 2610.

In this example, guide slice packets 2602 include intra-coded guide slices with PIDs 1-10 that are respectively associated with the ten IPG pages (g_1-g_{10}) shown in

FIG. 10C. Intra-coded video slice packets 2604 include intra-coded video slices with PID11, which correspond to the video picture (v_I) shown in FIG. 10C. In a preferred embodiment, audio packets 2606 with the audio PID are interleaved with guide slice packets 2602 and intra-coded video slice packets 2604 (e.g., as shown in FIG. 26) to form a guide/video/audio packet group 2612.

As shown in FIG. 26, data packets 2608 may follow guide/video/audio packet group 2612. Data packets 2608 may include, for example, data PIDs 1-10. Subsequently, following data packets 2608 are predictive slice packets 2610. Predictive slice packets 2610 include the predictive-coded slices with PID12, as shown in FIG. 10C.

Alternatively, the slices may be divided into multiple final transport streams in a manner analogous to that described above for FIGS. 23, 25A, and 25B. In addition, different program assignments may be imposed on each final transport stream to yield a single program or multiple programs in a manner analogous to that described above for FIGS. 21A and 21B.

The above examples are merely illustrative and not limiting. For example, the invention is not limited to embodiments with only ten IPG pages. Rather, the invention contemplates the use of any number of pages in the IPG, and ten pages are described only by way of illustration.

3. Latency Reduction

As described above in relation to the multiplexing structures, the IPG is preferably delivered using a single final transport stream. However, as the number of IPG pages increases, multiple final transport streams may be used depending on the bandwidth requirements of the elementary streams. When multiple transport streams are used, transitions between transport streams may have the undesired effect of introducing latencies (i.e., delays). The invention provides various methods to reduce switching latencies.

In a first method to reduce switching latencies between transport streams, related IPG pages are grouped into the same transport stream. Related IPG pages may be close in content, or close in time, or close in other relationship. Grouping related IPG pages advantageously provides for rapid changes between video PIDs within the same transport stream.

Grouping related IPG pages also enables the construction of relatively small transport streams that may be delivered in a targeted fashion to specific local neighborhoods and/or at specific times. Such targetable transport streams may be used to further reduce switching latencies.

For example, consider a first transport stream transmitting IPG pages for the next 1-hour of broadcast programming to a neighborhood. Suppose a viewer in the neighborhood wants to look ahead in the program listings to look at the following 1-hour of broadcast programming. Ordinarily, this may require a terminal to request the desired IPG pages from the head-end. However, in accordance with an embodiment of the invention, the latency of receiving such IPG pages may be reduced by the automatic transmission, along with the first transport stream, of a second transport stream for the IPG pages. This is advantageous in that the terminal needs not specifically request those IPG pages from the head-end.

FIG. 27A shows a second method to reduce switching latencies between transport streams. As shown in FIG. 27A, certain packets may be redundantly carried by more than one transport stream in order to reduce switching latencies. In the specific example illustrated in FIG. 27A, the video packets with PID3 are redundantly carried by both transport streams 2702 and 2704. Since the same video PID is included in two transport streams, a terminal can utilize either stream or both streams while transitioning from one transport stream to the other. In this manner, delays experienced by the viewer when the terminal changes from one transport stream to another are reduced because the transition may occur as a background process which does not interrupt the display.

The structure in which PIDs overlap between transport streams may be applied in various embodiments where multiple final transport streams are utilized. For example, the overlapping PID structure is applicable whether level zero, level one, or level two encoding is utilized. As a specific example, the slice-based single transport stream formation depicted in FIG. 26 may be extended to multiple slice-based transport streams with overlapping PIDs as described below.

FIG. 27B is a diagram illustrating slice-based multiple transport streams with overlapping PIDs to reduce latencies in accordance with an embodiment of the invention. In the example shown, each of transport streams 2752 and 2754 carries intra-coded guide slices identified by three PIDs. However, the three PIDs for the first

transport stream 2752 overlap with the three PIDs for the second transport stream 2754. In particular, each transport stream includes intra-coded guide slices identified by PID3.

The PID(s) to be shared between transport streams may be determined in various manners. In an embodiment, the IPG page that will most probably be used by a viewer to switch from one transport stream to another is determined or predetermined. For example, if the first transport stream can include pages listing broadcast programming and a page listing pay-per-view (PPV) movies, and the second transport stream can include pages enabling the ordering of PPV movies and related electronic commerce pages. The page listing PPV movies in the first transport stream may be predetermined to be the page most probably used by a viewer to switch from the first transport stream to the second transport stream. Hence, in accordance with an embodiment of the invention, the page listing PPV movies would be included in the first transport stream as well as the second transport stream, to efficiently and effectively reduce the latency in switching between the two transport streams.

It can be noted that each of the multiple transport streams described above may be structured as a single program or multiple programs. In an application where all the streams need to share the same time base, a single program is preferred. In other applications where the streams can have different time bases, multiple programs can be used whereby streams with similar time bases are grouped together and assigned to the same program.

FIG. 28 illustrates a third method for reducing switching latencies between transport streams. FIG. 28 shows an example IPG page with two threshold levels for stream priming in accordance with an embodiment of the invention. Stream priming is a method whereby a terminal anticipates that packets with particular PIDs may soon be needed and so requests those packets prior to the actual need for them.

For example, as shown in FIG. 28, switching from one IPG page to another may be anticipated using certain threshold settings in the guide portion of the IPG page. Consider a viewer traversing vertically within the page and passing an upper threshold (e.g., channel 18). Before the viewer selection reaches the end of the page, the terminal starts searching for the PIDs carrying the program guide for the next upper group of channels (e.g., channels 21-30). In accordance with an embodiment of the invention, if the current transport stream does not include those PIDs, then those PIDs are requested from the head-end once the threshold has been passed. The head-end then delivers those

PIDs, either in another transport stream, or by modifying the contents of the current transport stream. The delivery may be accomplished using either a pointcast to the requesting terminal or a narrowcast to a set of terminals that includes the requesting terminal. Analogous processes would occur when a viewer traverses vertically within the IPG page and passes a lower threshold.

The stream priming technique reduces latency by viewer user movement within a page to predict page switching beforehand and taking the appropriate action.

The stream priming technique may also be applied in a time dimension. For example, near the end of a particular 1-hour time period (e.g., within the last ½ hour of the period), the terminal may anticipate that a viewer may want to view the listings in the next 1-hour time period. Hence, if the current transport stream does not include the listings for the next time period, then the listings for the next time period are requested in anticipation of the demand.

4. Stream Indexing

In an embodiment, the head-end provides a program mapping table (PMT) for each broadcast channel. The PMT conveys to each terminal the PID assignment for each IPG (video, audio, and data) page being provided.

Consider, for example, a program guide including 24 time slots per day, with each time slot covering one hour. Further, consider a system with 20 IPG pages per time slot, with each IPG page assigned with a corresponding video PID. In this example, 24 slots x 20 PIDs per slot = 480 PIDs are required to provide program guide for one day. Also, if two weeks of programming content is to be stored at the head-end, then 14 days x 480 PIDs per day = 6720 PIDs are required for two weeks of program guide.

For each IPG page (e.g., each video PID), a data message can be used to deliver overlay, user interaction, and other desired features and functionality related to the page. This data may be delivered either using a separate data PID for each IPG page, or via a data PID that is shared by multiple IPG pages. The former option, however, may be impractical for a typical system. This is because if one data PID is needed for each IPG page, then the total number of PIDs needed to be stored at the head-end for two weeks doubles from 6720 to 13,440. Such a high number of PIDs are not currently supported by a typical encoding system. For example, MPEG-2 provides only 8192 PIDs for use due to its 13-bit PID, and some of those PIDs are pre-assigned or reserved.

FIG. 29 is a diagram illustrating a program mapping table (PMT) in accordance with an embodiment of the invention. The PMT includes a current programming area 2902 that contains, illustratively, 20 video PIDs, related data PIDs, and an audio PID for the 20 IPG pages covering the current 1-hour time slot (i.e., the time slot covering the programming currently being broadcast). Current programming area 2902 of the PMT is used (like a cache memory in some fashion) to temporarily store information that is most likely to be accessed by the viewers.

A next area 2904 of the PMT is allocated for the 2 weeks of video and audio programming to be stored. Illustratively, this area 2904 may include 6720 video and audio PIDs. Note that the current video and audio programming are also stored in this area 2904 (as well as in current programming area 2902).

A next area 2906 of the PMT is allocated for the 2 weeks of look-ahead data information associated with the look-ahead video information. For purposes of illustration, this look-ahead data area 2906 may be allocated 128 data PIDs, with each data PID being used to store look-ahead data information relating to multiple video PIDs.

Other areas of the PMT include areas reserved by MPEG-2 and areas reserved for future use.

FIGS. 30A and 30B are diagrams illustrating (a) prime time slots and (b) half-hour shifts of the current programming time slot, respectively, in accordance with an embodiment of the invention. As shown in FIG. 30A, the time periods in a day during which broadcast programming is most popularly watched are the three time slots between 5:00 pm (17:00) and 9:00 pm (21:00). In addition to such defined prime time period from 5:00 pm to 9:00 pm, the prime time information may be adjusted according to statistics of viewing on a local neighborhood or national scale.

As shown in FIG. 30B, the current programming time slot 3004 may be shifted in half-hour increments. While the 2 weeks of look-ahead IPG video data are stored in 1-hour time slots (e.g., 17:00 to 18:00, 18:00 to 19:00, and so on), the current programming time slot 3004 is arranged by half hour increments by retrieving and re-organizing the look-ahead video data as necessary.

FIG. 31 is a diagram illustrating a mapping of look-ahead video PIDs to look-ahead data PIDs in accordance with an embodiment of the invention. Such a mapping is used when there is substantially more look-ahead video PIDs (6720 in this example) than look-ahead data PIDs (128 in this example). When there is substantially

more video PIDs than data PIDs, each data PID is used on average to carry data information for multiple video PIDs. In this example, since there are 6720 look-ahead video PIDs and 128 look-ahead data PIDs, approximately 50 video PIDs are assigned on the average to each data PID. In particular, FIG. 31 illustrates, by way of example, the possible assignment of the first 50 look-ahead video PIDs to the first look-ahead data PID.

If the stream serving capability of the head-end were unlimited, then all 2 weeks of the look-ahead streams may be delivered from the head-end to the terminals. However, the limited stream serving capability of the head-end prevents this. In addition, it may not be necessary in practice to deliver all 2 weeks of the look-ahead streams because viewers do not typically require the guide information so far in advance. Hence, in accordance with an embodiment of the invention, only a subset of the 2 weeks of look-ahead streams may be delivered at any given moment in time.

FIG. 32 is a diagram illustrating television usage time during a typical week. As shown in FIG. 32, the usage typically peaks during the prime time period 3202 of a day. The daily pattern generally repeats itself during the weekdays, with non-prime time usage increasing on the weekends.

In addition to the general usage pattern with its weekly cycle illustrated in FIG. 32, certain IPG pages may receive particularly heavy viewing from certain viewer groups during certain time intervals. For example, the sport channel lists may receive particularly heavy viewing during the NBA (National Basketball Association) playoff games in the NBA playoff season. Hence, further evaluation of viewer IPG usage statistics may reveal other cyclic structures with different periods. These cyclic structures may be seasonal, as in the NBA playoff example.

These cyclic structures depend on, and may be characterized based on, common variables relating to the IPG system being used. These common variables may include, for example, t, p, and d. The variable t is a number from 1 to 24 representing a particular 1-hour time slot in a day. For example, the time slot from noon to 1:00 pm may be represented by t=13. The variable p is a number represents a particular IPG page among the total number of IPG pages (e.g., from 1 to 20). The variable d is a number from 1 to 14 representing a particular day of the 2 weeks of look-ahead programming (i.e., the number of look-ahead days).

FIG. 33A is a diagram illustrating a first look-ahead video PID layout 3300 in accordance with an embodiment of the invention. For each day, first video PID layout 3300 groups the 20 video PIDs for each time slot together, and further organizes the groups serially in ascending order of the variable t, going from t=1 to t=24. Further, first layout 3300 serially repeats the daily organization for each of the 14 days, going from d=1 to d=14.

Based on first look-ahead video PID layout 3300, daily prime time viewings follow each other in a cycle with a periodicity of 480 PIDs (the number of video PIDs for a day). This periodicity corresponds to incrementing the variable d by one.

Other possible viewing cycles may have different periodicities in terms of the variables p, t, and d. For example, a very popular show broadcast every Monday at 9:00 PM (in time slot t=21) may have its corresponding IPG page (e.g., page p=17) viewed very frequently. This would relate to a viewing cycle for page p=17 at time slot t=21 which repeats in increments of 7 for variable d. Hence, many viewing cycles may be characterized in terms of periodicities in the variables p, t, and d.

It may be undesirable to map many very popularly viewed video PIDs on the same data PID because of the uneven load distribution this may cause. Instead, it is advantageous to distribute the popularly viewed video PIDs evenly among the data PIDs to balance the load. One algorithm for such distribution is described below.

FIG. 33B is a diagram illustrating a method 3320 of forming a second look-ahead video PID layout in accordance with an embodiment of the invention. Method 3320 of forming the second layout includes two steps. The first step 3322 involves choosing the largest prime number that is less than or equal to the number of look-ahead data PIDs available. In this example, the number of look-ahead data PIDs available is 128, so the prime number within that constraint is 127.

The second step 3324 involves assigning a data PID to each video PID. This is done by taking the video PID number and performing a modulo with the prime number. Equivalently, the video PID number is divided by the prime number and the remainder of that division is the data PID number to be assigned to the video PID. For example, if the video PID number is 260, then data PID number 6 is assigned.

Method 3320 of FIG. 33B results in uniform distribution among the data PIDs of extensively viewed video PIDs with various cyclic periods. The uniform distribution results because a prime number does not contain any multiples of any other

number, so a periodic sequence of numbers divided by a prime number yields a different remainder for each entry in the sequence.

For example, consider the following cyclic sequence of video PIDs with a periodicity of 480: 0, 480, 960, and so on. Dividing each entry in the sequence by the prime number 127 yields the following remainders: 0, 99, 71, and so on. This sequence of remainders becomes the data PIDs assigned to the corresponding video PIDs. Notice that the assigned data PID is generally not repeated using this method. In this way, method 3320 achieves even distribution among data PIDs of extensively viewed video PIDs with various cyclic periods.

Alternatively, if the divisor selected is not a prime number, then the distribution may be uneven. For example, if the divisor is 120, then for the above cyclic sequence of video PIDs with periodicity of 480, dividing by 120 yields the following remainders: 0, 0, 0, 0, and so on. Hence, in this example, each of the video PIDs in the sequence would be assigned to the same data PID (e.g., data PID0). If all those video PIDs were for prime time, then data PID0 would receive a large and uneven load of usage.

FIG. 33C is a diagram illustrating the distribution of data messages among data PIDs in accordance with an embodiment of the invention. FIG. 33C relates to the case where multiple data messages (associated with multiple video PIDs) share the same data PID.

In FIG. 33C, the small "d" represents non-prime time data messages, and the capital "D" represents prime time data messages. Due to the application of method 3320 of FIG. 33B to determine assignment of the data messages to the data PIDs, the prime time data messages D are evenly distributed among the data PIDs.

G. SYSTEM

1. Head-End

FIG. 12A is a block diagram of an embodiment of an information distribution system 1200 that can be used to provide interactive program guide and to implement various aspects of the invention. Distribution system 1200 includes a head-end 1202, local neighborhood equipment (LNE) 1204, one or more distribution nodes 1206 (e.g., a hybrid fiber-coax network), and a number of set top terminals (STTs) 1208.

Distribution system 1200 is described in further detail in U.S. Patent Application Serial No. 08/984,710, filed December 3, 1997; Serial No. **(Attorney Docket No. 19880-002300)**, entitled "SERVICE PROVIDER SIDE IPG ENCODER," filed November 1, 1999; Serial No. **(Attorney Docket No. 19880-001630)**, entitled "MESSAGING PROTOCOL FOR DEMAND-CAST SYSTEM AND BANDWIDTH MANAGEMENT," filed March 30, 2000; and Serial No. **(Attorney Docket No. 19880-001210)**, entitled "SYSTEM AND METHOD FOR DELIVERY OF SHORT-TIME DURATION VIDEO SEGMENTS," filed June 27, 2000. These patent applications are assigned to the assignee of the invention and incorporated herein by reference. One specific implementation of distribution system 1200 is known as the DIVA™ System provided by DIVA Systems Corporation.

Head-end 1202 produces a number of digital streams that contain encoded information in (e.g., MPEG-2) compressed format. These streams are then modulated using a modulation technique that is compatible with a communications channel 1262 that couples head-end 1202 to one or more LNEs 1204 (only one LNE 1204 is shown in FIG. 12A for simplicity). LNE 1204 is typically located away from head-end 1202. LNE 1204 selects data for viewers in the LNE's neighborhood and re-modulates the selected data in a format that is compatible with distribution node 1206. Although system 1200 is depicted as having head-end 1202 and LNE 1204 as separate components, those skilled in the art can realize that the functions of the LNE may be incorporated into head-end 1202. Also, the elements of system 1200 can be physically located anywhere, and need not be near each other.

In system 1200, the program streams are addressed to particular STT locations that requested the information through an interactive menu. An interactive menu structure for requesting video-on-demand is disclosed in commonly assigned U.S. Patent Application Serial No. 08/984,427, filed December 3, 1997. Another example of the interactive menu for requesting multimedia services is the interactive program guide disclosed in commonly assigned U.S. Patent Application Serial No. 60/093,891, filed in July 23, 1998.

To assist a viewer in selecting programming, head-end 1202 produces information that can be assembled to create an IPG page such as that shown in FIG. 9. Head-end 1202 produces the components of the IPG page as bitstreams that are compressed prior to transmission.

Within head-end 1202, a video source 1212 supplies a video sequence for the video portion of the IPG pages, an audio source 1214 supplies one or more audio signals associated with the video sequence, and a guide data source 1216 provides program guide data for the guide portion of the IPG pages. The guide data is typically in a database format, where each entry describes a particular program by its title, presentation time, presentation date, descriptive information, channel, and program source. The video sequence, audio signals, and program guide data are provided to an encoder unit 1210.

Encoder unit 1210 (which is described in further detail below) compresses the received video sequence into one or more elementary streams, the audio signals into one or more elementary streams, and the guide produced from the guide data into one or more elementary streams. The elementary streams can be produced using a picture-based encoding technique, a slice-based encoding technique, or a combination thereof, as described above. The elementary streams are then provided to an in-band delivery system 1250 (e.g., cable modem).

Within delivery system 1250, the elementary streams are assembled into one or more transport streams that are then modulated using a modulation format that is compatible with communication channel 1262. For example, communication channel 1262 may be a fiber optic channel that carries high-speed data from head-end 1202 to a number of LNE 1204. LNE 1204 selects the IPG page components that are applicable to its neighborhood and re-modulates the selected data into a format that is compatible with distribution node 1206. A detailed description of LNE 1204 is described in U.S. Patent Application Serial No. 09/583,388, entitled "ENCODING OPTIMIZATION TECHNIQUES FOR ENCODING PROGRAM GRID SECTIONS OF SERVER-CENTRIC INTERACTIVE PROGRAM GUIDE," filed May 30, 2000, assigned to the assignee of the invention and incorporated herein by reference.

STT 1208 receives and demodulates the signals provided by distribution node 1206 and decodes the demodulated signals to retrieve the IPG pages from the stream. The design of STT 1208 is described in further detail below.

As shown in FIG. 12A, encoder unit 1210 includes a video processor 1220 and a graphics processor 1240. Video processor 1220 further includes a compositor unit 1222 and an encoder 1224. Compositor unit 1222 combines the video sequence from video source 1212 with advertising video, advertiser or service provider logos, still

graphics, animation, other video information, or a combination thereof. The video sequence from compositor unit 1222 is then provided to encoder 1224.

Encoder 1224 includes one or more video encoders 1226 (e.g., real-time MPEG-2 encoders) and one or more audio encoders 1228 (e.g., AC-3 encoders). Video encoder 1226 receives the video sequence from compositor unit 1222 and forms a (e.g., slice-based) bitstream (e.g., an MPEG-2 compliant bit stream) for the video portion of an IPG page. In an embodiment, video encoder 1226 "pads" the graphics portion (illustratively the left half portion of the IPG page corresponding to the guide listing) with null data. The null data may be replaced by the graphics grid slices (e.g., at a later step, within the LNE). In this embodiment, video encoder 1226 is designed for, and efficiently processes only motion video information, excluding the graphics data. Audio encoder 1228 receives the audio signals and forms a bitstream for the audio portion of the IPG page. Encoder 1224 produces one or more elementary streams containing picture-based or slice-based encoded video and audio information.

A controller 1230 couples to encoder unit 410 and manages the (e.g., slice-based) encoding process such that the video encoding process is temporally and spatially synchronized with the grid encoding process. For slice-based encoding, this synchronization can be achieved by defining the slice start and stop locations according to the objects in the IPG page layout and managing the encoding process as defined by the slices.

In an embodiment, the graphics (e.g., guide) portion of the IPG page is separately encoded by graphics processor 1240. Graphics processor 1240 receives the guide data from guide data source 1216. A guide data grid generator 1242 within graphics processor 1240 formats the guide data into a "grid", e.g., having a vertical axis of program sources and a horizontal axis of time increments. The guide grid is a video picture that is encoded using a guide encoder 1244 designed for video with text and graphics content. Guide encoder 1244, which can be implemented in software, encodes the guide data grid (e.g., via a slice-based encoding technique) to produce one or more bitstreams that collectively represent the entire guide data grid. Guide encoder 1244 is designed to effectively encode the graphics and text content.

For slice-based encoding, controller 1230 defines the start and stop macroblock locations for each slice. The result is a GOP structure having intra-coded pictures containing intra-coded slices and predicted pictures containing predictive-coded

slices. The intra-coded slices are separated from the predictive-coded slices. Each coded slice is separately stored in a slice-form grid page database 1246. The individual slices can be addressed and retrieved from database 1246 as required for transmission. Controller 1230 controls the slice-based encoding process and further manages database 1246.

For a server-centric system, since the program guide database resides at the head-end, a two-way communication system via a back-channel 1264 from terminal 1208 through distribution node 1206 to head-end 1202, is utilized to support requests from the terminal. Back-channel 1264 can be used to send requests and other messages from terminal 1208 to head-end 1202.

2. Local Neighborhood Equipment (LNE)

FIG. 12B is a block diagram of an embodiment of LNE 1204. In this embodiment, LNE 1204 includes a cable modem 1272, slice combiner 1274, a multiplexer 1276 and a digital video modulator 1278. LNE 1204 is coupled illustratively via cable modem 1272 to head-end 1202 and receives one or more transport streams containing the encoded video, guide, data, and audio information. Cable modem 1272 demodulates the signal from head-end 1202 and extracts the (MPEG) coded information from the received signal. Slice combiner 1274 combines the received video slices with the guide slices in an order such that the decoder at the terminals can easily decode the IPG without further slice re-organization. The resultant combined slices are assigned PIDs and formed into one or more (e.g., MPEG-compliant) transport streams by multiplexer 1276. The scanning, combination, and multiplexing of the slices are described above. The transport stream(s) are transmitted via a digital video modulator 1278 to distribution node 1206.

LNE 1204 is programmed to extract particular information from the signal transmitted by head-end 1202. As such, LNE 1204 can extract video and guide slices that are targeted to the viewers coupled to the LNE. For example, LNE 1204 can extract specific channels for representation in the guide grid that are available to the viewers coupled to that LNE. As such, unavailable channels to a particular neighborhood would not be depicted in a viewer's IPG. Additionally, the IPG can include targeted advertising, e-commerce, program notes, and others. As such, each LNE can combine different guide slices with different video slices to produce IPG pages that are prepared specifically for

the viewers coupled to that particular LNE. Other LNEs may select different IPG component information that is relevant for their associated viewers.

3. Set Top Terminal

FIG. 34 depicts a block diagram of an embodiment of set top terminal (STT) 3408 suitable for producing an IPG page and supporting various aspects of the invention. STT 3408 includes a tuner 3412, a demodulator 3414, a transport demultiplexer 3418, an audio decoder 3420, a video decoder 3430, an on-screen display (OSD) processor 3432, a video compositor 3434, a frame store memory 3436, a controller 3450, and a modulator 3470. User interaction is provided via a remote control unit 3480. Tuner 3412 receives, e.g., a radio frequency (RF) signal comprising, for example, a number of broadcast (e.g., QAM) signals from a downstream (forward) channel. Tuner 3412, in response to a control signal TUNE, tunes to and processes a particular broadcast signal to produce an intermediate frequency (IF) signal. Demodulator 3414 receives and demodulates the IF signal to produce an information stream, illustratively an MPEG transport stream. The transport stream is provided to a transport stream demultiplexer 3418.

Demultiplexer 3418, in response to a control signal TD produced by controller 3450, demultiplexes (i.e., extracts) an audio stream A and a video stream V. The audio stream A is provided to audio decoder 3420, which decodes the audio stream and provides a decoded audio stream to an audio processor (not shown) for subsequent presentation. The video stream V is provided to video decoder 3430, which decodes the compressed video stream V to produce an uncompressed video stream VD that is provided to video compositor 3434. OSD processor 3432, in response to a control signal OSD produced by controller 3450, produces a graphical overlay signal VOSD that is provided to video compositor 3434. In an embodiment, during transitions between streams representing different IPG pages, the buffers in the decoder are not reset. As such, the pages seamlessly transition from one page to another.

Video compositor 3434 merges the graphical overlay signal VOSD and the uncompressed video stream VD to produce a modified video stream (i.e., the underlying video images with the graphical overlay) that is provided to frame store unit 3436. Frame store unit 3436 stores the modified video stream on a frame-by-frame basis according to the frame rate of the video stream. Frame store unit 3436 provides the stored video

frames to a video processor (not shown) for subsequent processing and presentation on a display device.

Controller 3450 includes an input/output module 3452, a microprocessor 3454, support circuitry 3456, an infrared (IR) receiver 3458, and a memory 3460. Input/output module 3452 forms an interface between controller 3450 and tuner 3412, transport demultiplexer 3418, OSD processor 3432, back-channel modulator 3470, and remote control unit 3480. Microprocessor 3454 cooperates with support circuitry 3456 such as power supplies, clock circuits, cache memory, and the like as well as circuits that assist in executing the software routines that are stored in memory 3460.

Although controller 3450 is depicted as a general-purpose processor that is programmed to perform specific interactive program guide control function in accordance with the invention, the controller can be implemented in hardware as an application specific integrated circuit (ASIC). As such, the process steps described herein are intended to be broadly interpreted as being equivalently performed by software, hardware, or a combination thereof.

In the embodiment shown in FIG. 34, remote control unit 3480 includes an 8-position joystick, a numeric pad, a "Select" key, a "Freeze" key and a "Return" key. User manipulations of the joystick or keys of the remote control device are transmitted to controller 3450 via an infrared (IR) link or an RF link. Controller 3450 is responsive to such user manipulations, executes related user interaction routines 3462, and uses particular overlays that are available in an overlay storage 3466.

After the signal is tuned and demodulated, the video streams are recombined via a stream processing routine 3468 to form the video sequences that were originally compressed. Stream processing routine 3468 employs a variety of methods to recombine slice-based streams, including using PID filter 3416 and demultiplexer 3418, as described in the aforementioned U.S. Patent Application Serial No. 09/583,388. Note that the PID filter implemented illustratively as part of demodulator 3414 is utilized to filter the undesired PIDs and retrieve the desired PIDs from the transport stream. The packets to be extracted and decoded to form a particular IPG page are identified by a PID mapping table 3464. After stream processing routine 3468 has processed the streams into the correct order (assuming the correct order was not produced in the LNE), the slices are sent to (MPEG) video decoder 3430 to generate the original uncompressed IPG pages.

If a transport stream with two PIDs as described above is to be received and processed (e.g., for slice-based decoding), stream processing unit 3468 recombines the intra-coded slices with their corresponding predictive-coded slices in the appropriate order before the recombined streams are coupled to video decoder 3430. This process can be implemented by software or hardware, or a combination thereof. In the slice structure, only one slice is assigned per row and each row is divided into two portions (e.g., the guide portion and the video portion). In order for the receiving terminal to reconstruct the original video picture, one method is to construct the first row from its two slices in the correct order by retrieving two corresponding slices from the transport stream, then construct the second row from its two slices, and so on. In this manner, the terminal processes two PIDs in the same time period.

PID filter 3416 can be programmed to pass the desired PIDs and filter out the undesired PIDs. The desired PIDs are identified by controller 3450 after the viewer selects particular IPG page to review. PID mapping table 3464 is accessed by controller 3450 to identify which PIDs are associated with the desired IPG. If PID filter 3416 is available in the receiver terminal, it is used to retrieve the PIDs containing slices for the guide and video portions. Demultiplexer 3418 then extracts packets from these PIDs and provides the packets to video decoder 3430, in the order in which they arrived. If the STT does not have optional PID filter 3416, then demultiplexer 3418 performs the PID filtering and extracting functions. Depending on the particular STT implementation, a corresponding method is used to recombine and decode slice-based streams. These various methods are described in further detail below and in the aforementioned U.S. Patent Application Serial No. 09/583,388.

H. RECOMBINATION METHOD FOR SLICE-BASED DECODING

The transmitted slices for the IPG pages, encoded in the manner described above, can be recombined in various manners. Some of these recombination methods are described below.

1. First Recombination Method

In the first recombination method, the slice-based intra-coded streams (e.g., for the guide and video portions) and the slice-based predictive-coded streams (for the predictive-coded pictures) to be recombined keep their separate PIDs until the point

where they are depacketized. The recombination process is conducted within the transport demultiplexer of the terminal. For illustrative purposes, in a multi-program transport stream, each program consists of an I-PID for each intra-coded guide portion, one or more I-PIDs for the intra-coded video portion, a predictive PID for the predictive-coded guide and video portions, an audio PID, and a number of data PIDs. Any packet with a PID that matches any of the PIDs within the desired program (as identified in a program mapping table) are depacketized and the payload is sent to the video decoder. Payloads are sent to the decoder in the order in which the packets arrive at the demultiplexer.

FIG. 35 is a flow diagram of an embodiment of a first recombination process 3500. At step 3510, the process waits for a (viewer) selection for a picture (e.g., a particular IPG page) to be received. The I-PID for the selected picture, as the first picture of a video stream's GOP, identifies the stream to be received. However, since the slice-based encoding technique assigns two or more I-PIDs to the stream (i.e., an I-PID for the guide portion and one or more I-PIDs for the video portion), all (two or more) I-PIDs assigned for the selected picture are identified. A packet having any one of the identified I-PIDs is then detected.

At step 3515, the I-PID packets (e.g., packets with PID1 and PID11 for IPG page 1 in FIG. 10C) are extracted from the transport stream, including the header information and data, until the next picture start code. The header information within the first received I-PID access unit includes a sequence header, a sequence extension, a group start code, a GOP header, a picture header, and a picture extension, which are known to a reader that is skilled in MPEG-1 and MPEG-2 compression standards. The header information in the next I-PID access unit that belongs to the second and later GOPs includes the group start code, the picture start code, the picture header, and an extension. At step 3520, the payloads of the packets that include header information related to the video stream and the intra-coded picture are coupled to the video decoder as video information stream V.

At step 3525, the slice-based predictive-coded packets PRED-PID (e.g., PID12 in FIG. 10C) for fourteen predictive-coded pictures in a GOP of size fifteen are extracted from the transport stream. At step 3530, the payloads of the packets that include the header information related to the video stream and the predicted-coded pictures are coupled to the video decoder as video information stream V. At the end of

step 3530, a complete GOP, including the intra-coded and predictive-coded slices, are available to the video decoder. As the payloads are sent to the decoder in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination processing.

At step 3535, a query is then made whether a different picture is requested, (e.g., a new IPG is selected). If a different picture is not requested, then the process returns to step 3510 and the demultiplexer waits for the next packets having the PIDs of the desired I-PIDs. Otherwise, if a different picture is requested, then the I-PIDs of the new desired picture are identified at step 3540, and the process returns to step 3510.

The process shown in FIG. 35 can be used to produce an MPEG-compliant video stream V by recombining the desired intra-coded slices and the predictive-coded slices from the GOP structure.

2. Second Recombination Method

In the second method for recombining the video stream, the transport stream is modified using a PID filter. The PID filter can be implemented as part of the demodulator, as shown in FIG. 34, or as part of the demultiplexer.

For illustrative purposes, in a multi-program transport stream, each program can include a number of I-PIDs for the video and guide portions, a predictive PID for the video and guide portions, an audio PID, and a number of data PIDs. Any packet with a PID that matches any of the PIDs in the desired program, as identified by the program mapping table (PMT) has its PID modified to the lowest PID in the program (the PID that is referenced first in the program's PMT). As a specific example, a program can include a guide slice I-PID of 50, a video slice I-PID of 51, and a predictive PID of 52. For this program, the PID-filter modifies the video I-PID and the predictive PID to 50 and thereby, the intra-coded and predictive-coded access units attain the same PID number and become a portion of a common stream. As a result, the transport stream from the PID filter contains a program with a single video stream having packets that appear in the proper order to be decoded as valid MPEG bitstream.

Note that the incoming bit stream does not necessarily contain any packets with a PID equal to the lowest PID referenced in the program's PMT. Also note that it is possible to modify the PIDs to other PID numbers than lowest PID without changing the operation of the process.

When the PIDs of incoming packets are modified to match the PIDs of other packets in the transport stream, the continuity counters of the merged PIDs may become invalid at the merge points, since each PID has its own continuity counter. For this reason, the discontinuity indicator in the adaptation field is set for any packets that may immediately follow a merge point. Any decoder components that check the continuity counter for continuity properly processes the discontinuity indicator bit.

FIG. 36 is a flow diagram of an embodiment of a second recombination process 3600. At step 3610, the process waits for a (viewer) selection of two I-PIDs (e.g., two PIDs corresponding to the guide and video slices) to be received. The I-PIDs, comprising the first picture of a video stream's GOP, identify the two streams to be received. A packet having any one of the selected I-PIDs is then detected.

At step 3615, the PIDs of the intra-coded guide and video portions are re-mapped to a particular number (e.g., PID*). At this step, the PID filter modifies all PIDs of the desired I-stream packets to PID*. At step 3620, the PID number of the predictive-coded pictures (predictive PID) is also re-mapped to PID* by the PID filter, which modifies all PIDs of the predictive PID packets to PID*.

At step 3625, the packets of the PID* stream are extracted from the transport stream by the demultiplexer. At step 3630, the payloads of the packets that includes the video stream header information and the intra-coded and predictive-coded slices are coupled to the video decoder as video information stream V. It should be noted that the slice packets are ordered in the transport stream in the same order as they are to be decoded (e.g., the guide slice packets for first row followed by the video slice packets for first row, then the slices for the second row, and so on).

At step 3635, a query is made whether a different picture (e.g., another IPG page) is requested. If a different picture is not requested, then the process returns to step 3610 where the demultiplexer waits for the next packets having the identified I-PIDs. Otherwise, if a different picture is requested, then the I-PIDs of the new desired picture are identified at step 3640 and the process returns to step 3610.

The process shown in FIG. 36 is used to produce an MPEG-compliant video stream by merging the intra-coded slices and predictive-coded slices before the demultiplexing process.

3. Third Recombination Method

The third recombination method accomplishes MPEG bitstream recombination by using splicing information in the adaptation field of the transport packet headers and by switching between video PIDs based on splice countdown concept.

In the third recombination method, the MPEG streams signal the PID-to-PID switch points using the splice countdown field in the transport packet header's adaptation field. When the PID filter is programmed to receive one of the PIDs in a program's PMT, the reception of a packet containing a splice countdown value of 0 in its header's adaptation field causes immediate reprogramming of the PID filter to receive another video PID. It should be noted that special attention to splicing syntax is required for systems that use splicing for other purposes.

FIG. 37 is a flow diagram of an embodiment of a third recombination process 3700. At step 3710, the process waits for a (viewer) selection of the I-PIDs to be received for the desired IPG page. The I-PIDs, comprising the first picture of a stream's GOP, identify the stream to be received. A packet having any one of the selected I-PIDs is then detected .

At step 3715, the I-PID packets are extracted from the transport stream until, and including, the I-PID packet with a slice countdown value of zero. At step 3720, the payloads of the packets that include the header information related to the video stream and the intra-coded slices are coupled to the video decoder as video information stream V.

At step 3725, the PID filter is re-programmed to receive the predictive-coded pictures. At step 3730, the predictive-coded packets (e.g., PID12 packets in FIG. 10C) are extracted from the transport stream. At step 3735, the payloads of the packets that include the header information related to the video stream and the predictive-coded pictures are coupled to the video decoder. At the end of step 3735, a complete GOP, including the intra-coded slices and the predictive-coded slices, are available to the video decoder. As the payloads are sent to the video decoder in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination processing.

At step 3740, a query is made whether a different picture (e.g., another IPG page) is requested. If a different picture is not requested, the process proceeds to step 3750 where the PID filter is re-programmed to receive the previous desired I-PIDs.

Otherwise, if a different picture is requested, then the I-PIDs of the new desired picture are identified at step 3745 and the process proceeds to step 3750 where the PID filter is re-programmed to receive the new I-PIDs. The process then returns to step 3710, where the demultiplexer waits for the next packets having the PIDs of the desired picture.

The process shown in FIG. 37 can be used to produce an MPEG-compliant video stream, where the PID-to-PID switch is performed based on a splice countdown concept. It should be noted that the slice recombination can also be performed using the second recombination method whereby the demultiplexer receives the PIDs and extracts packets from the transport stream based on the splice countdown concept. In this case, the same process is applied as shown in FIG. 37 with the difference that, instead of reprogramming the PID filter after the "0" splice countdown packet, the demultiplexer is programmed to depacketize the desired PIDs.

4. Fourth Recombination Method

For terminals that do not include a PID filter and for those in which the demultiplexer cannot process two PIDs for splicing the streams, a fourth recombination method described below can be used for stream recombination. In a terminal not capable of processing two PIDs, two or more streams with different PIDs are spliced together via an additional splicing software or hardware and can be implemented as part of the demultiplexer. In the fourth recombination method, information about which PID to be spliced as the next step is provided to the demultiplexer. The demultiplexer then processes only one PID, but a different PID after the splice occurs.

FIG. 38 is a flow diagram of an embodiment of a fourth recombination process 3800 for recombining the IPG streams. At step 3802, the process defines an array of elements having a size that is equal to the number of expected PIDs to be spliced. It is possible to distribute splice information in a picture as desired according to the slice structure of the picture and the desired processing form at the terminal. For example, in the slice-based streams described above, for an I-picture, splice information may be inserted into slice row portions of the guide and video data. At step 3804, the process initializes the video PID hardware for each entry in the array. At step 3810, the hardware splice process is enabled and the packets are extracted by the demultiplexer. The packet extraction may also be performed at another step within the demultiplexer. At step 3812, the process checks a hardware register to determine if a splice has been completed. If the

splice has occurred, the process disables the splice hardware, at step 3814, and sets the video PID hardware to the next entry in the array, at step 3816. The process then returns to step 3810. If the splice has not occurred, the process proceeds to step 3820, waits for a period of time, and then returns to step 3812.

In the above-described manner, the slices are spliced together by the hardware within the terminal. To facilitate recombination of the slices, the terminal is sent an array of valid PID values for recombining the slices via a user data in the transport stream or another communications link between the terminal and the head-end. The array is updated dynamically to ensure that the correct portions of the IPG are presented to the viewer correctly. Since the splice points in the slice-based streams may occur at a frequent level, a software application may not have the capability to control the hardware for splicing operation as discussed above. In such case, a firmware may be dedicated to control the demodulator hardware at a higher rate for the splicing process.

The foregoing description of the preferred embodiments is provided to enable any person skilled in the art to make or use the invention. Various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments without the use of the inventive faculty. Thus, the invention is not intended to be limited to the embodiments shown herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

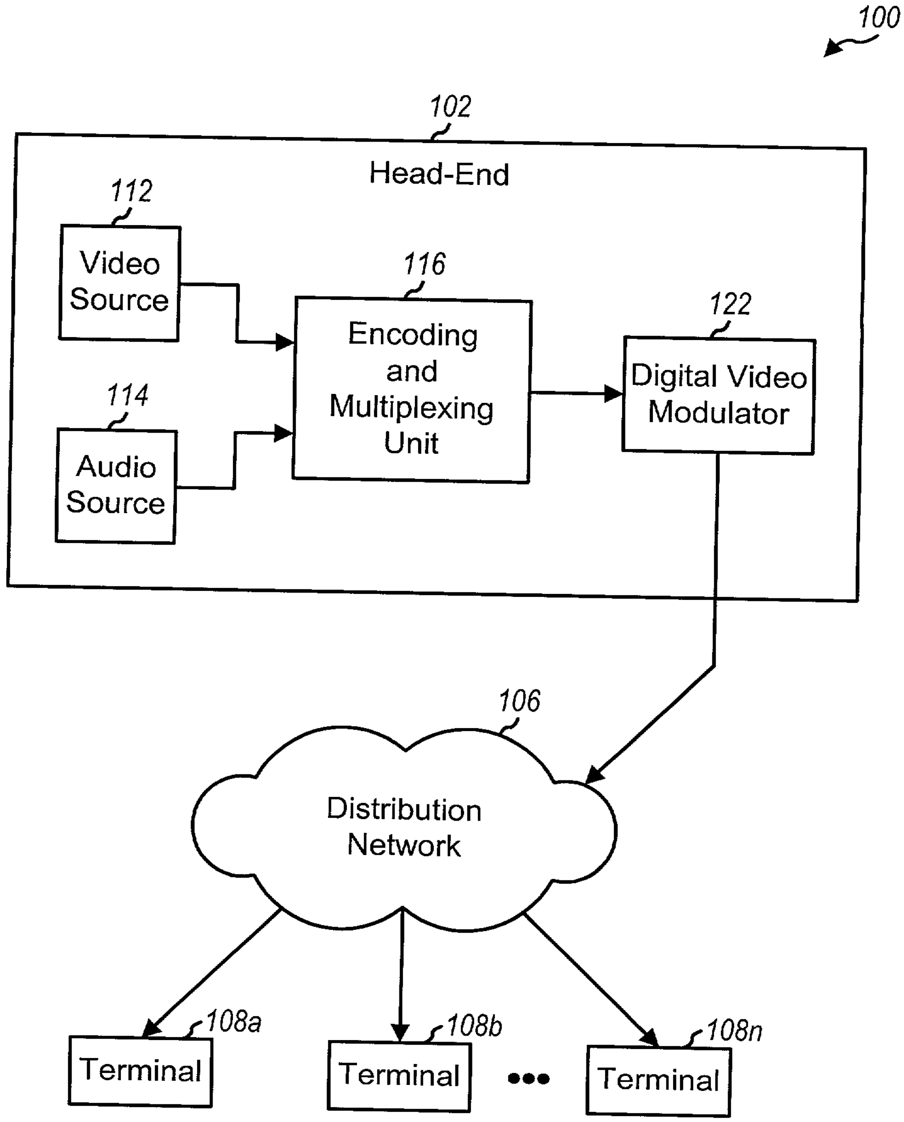

100
102
Head-End
112
Video Source
114
Audio Source
116
Encoding and Multiplexing Unit
122
Digital Video Modulator
106
Distribution Network
108a
Terminal
108b
Terminal
108n
Terminal

FIG. 1

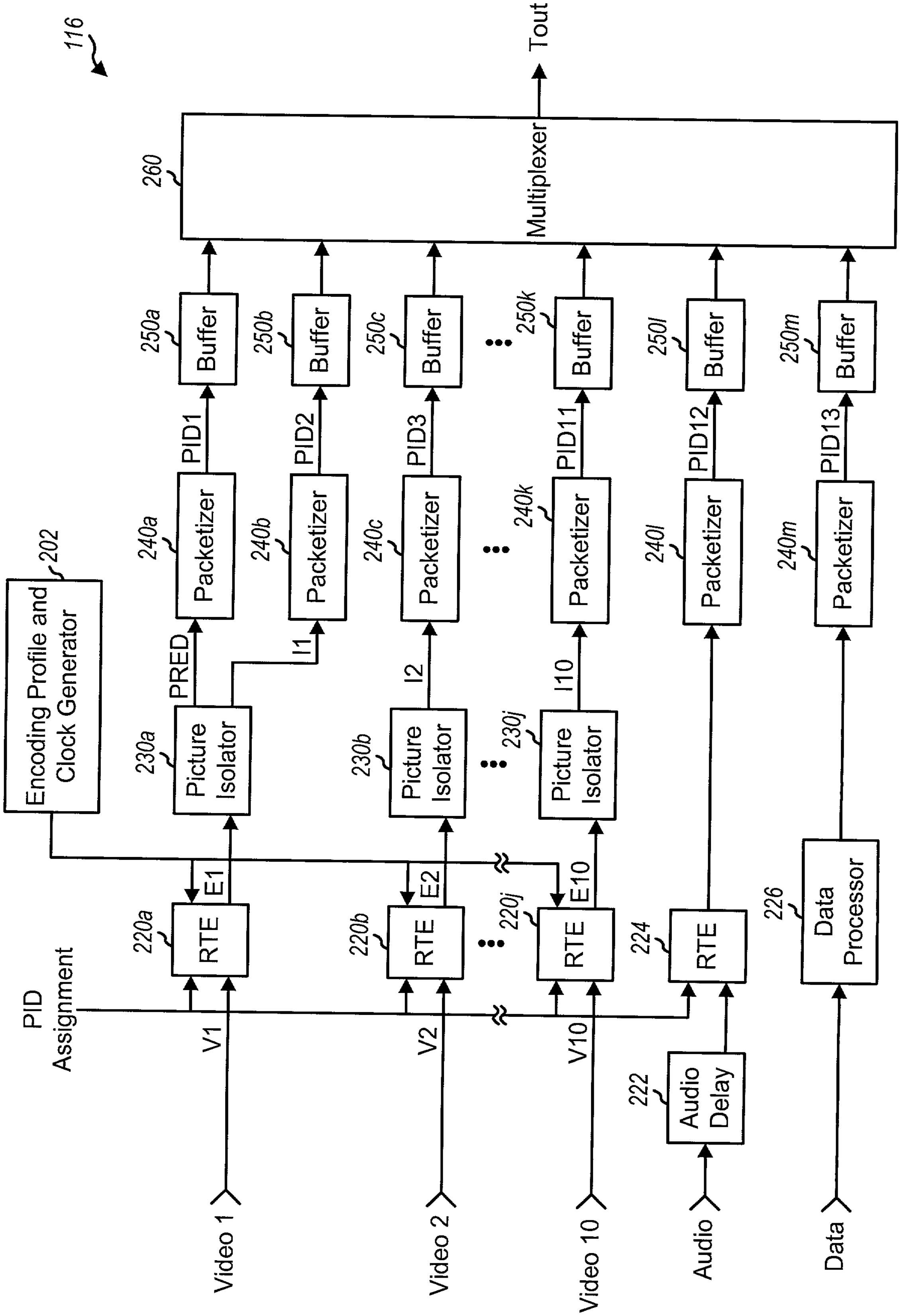

116
Tout
260
Multiplexer
250a
Buffer
250b
Buffer
250c
Buffer
250k
Buffer
250l
Buffer
250m
Buffer
PID1
PID2
PID3
PID11
PID12
PID13
240a
Packetizer
240b
Packetizer
240c
Packetizer
240k
Packetizer
240l
Packetizer
240m
Packetizer
202
Encoding Profile and Clock Generator
PRED
I1
I2
I10
230a
Picture Isolator
230b
Picture Isolator
230j
Picture Isolator
E1
E2
E10
220a
RTE
220b
RTE
220j
RTE
224
RTE
226
Data Processor
PID Assignment
V1
V2
V10
222
Audio Delay
Video 1
Video 2
Video 10
Audio
Data

FIG. 2

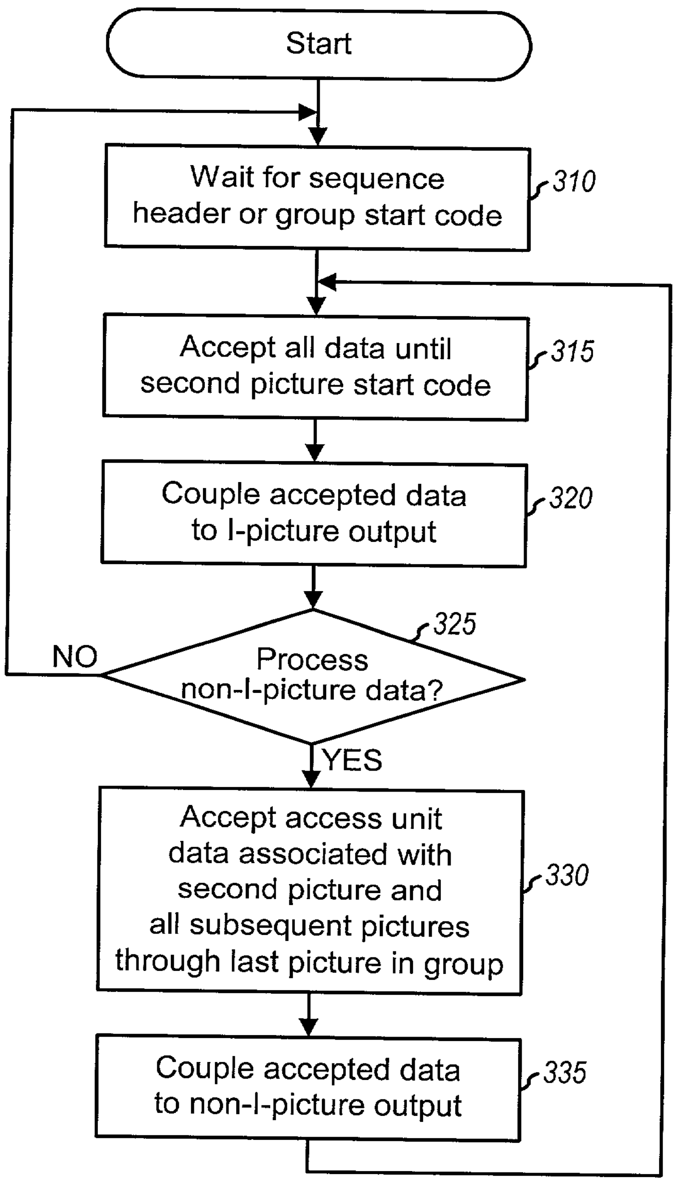

Start
Wait for sequence header or group start code
310
Accept all data until second picture start code
315
Couple accepted data to I-picture output
320
325
NO
Process non-I-picture data?
YES
Accept access unit data associated with second picture and all subsequent pictures through last picture in group
330
Couple accepted data to non-I-picture output
335

FIG. 3

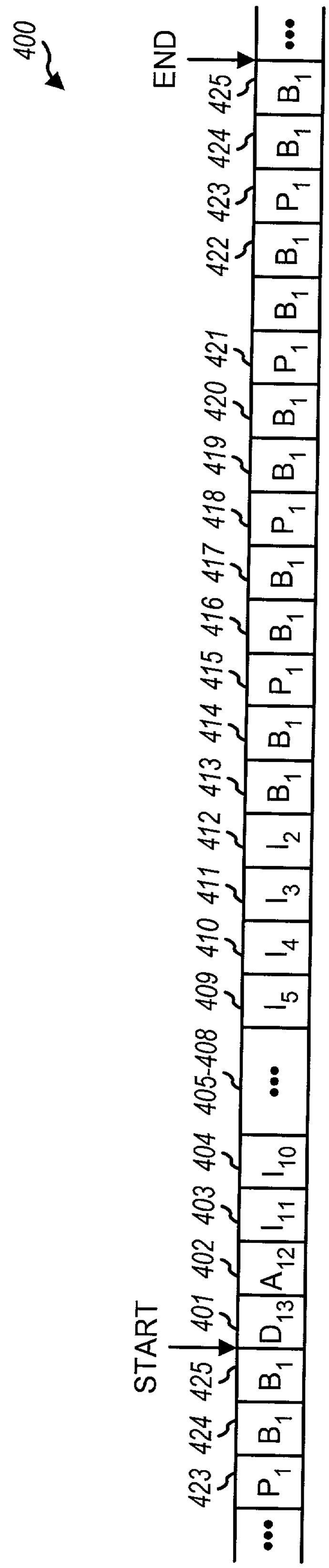

400
START
END
423
424
425
401
402
403
404
405-408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
P1
B1
B1
D13
A12
I11
I10
I5
I4
I3
I2
B1
B1
P1
B1
B1
P1
B1
B1
P1
B1
B1
P1
B1
B1

FIG. 4

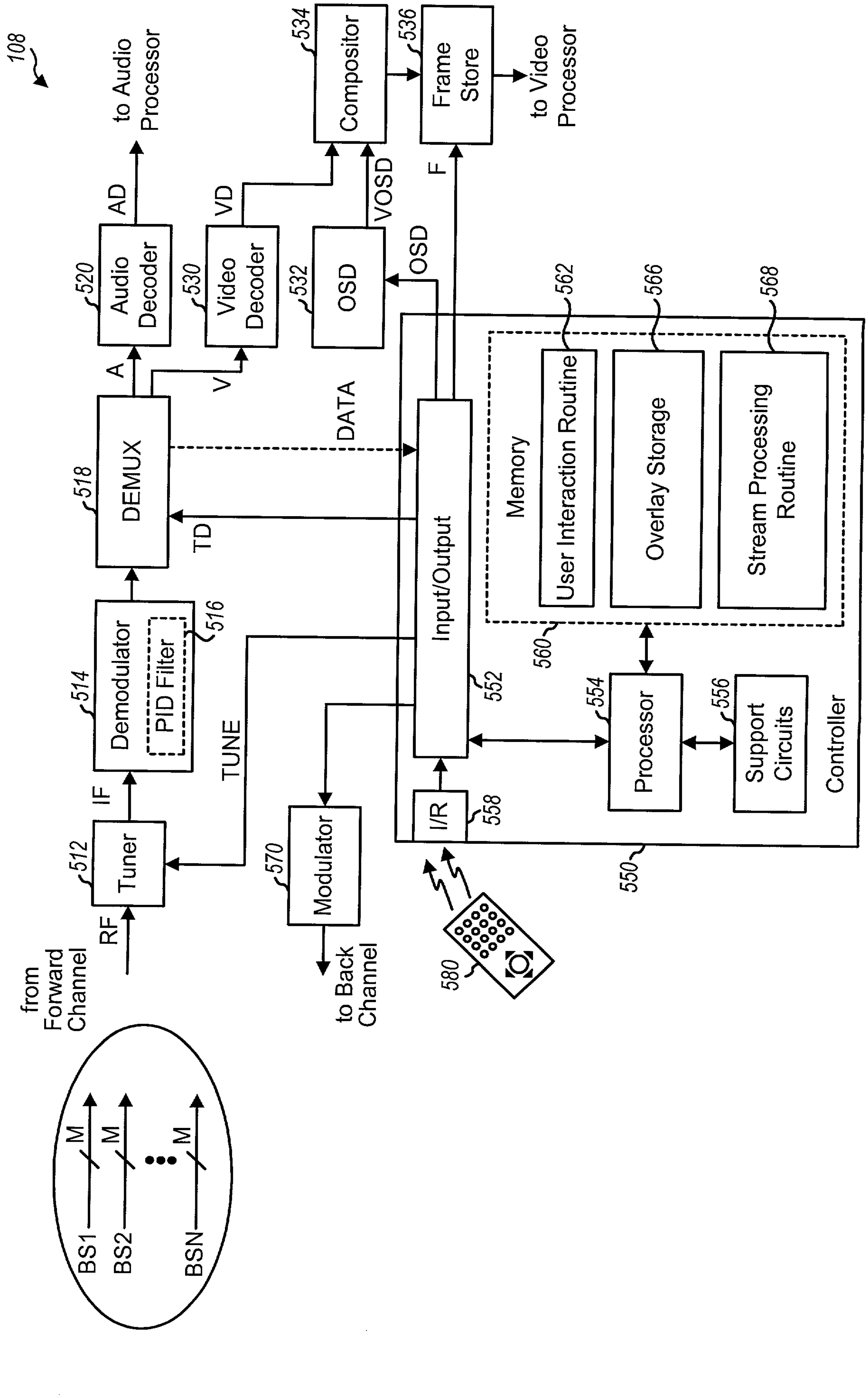

108
to Audio Processor
AD
520 Audio Decoder
A
530 Video Decoder
V
VD
534 Compositor
VOSD
536 Frame Store
to Video Processor
F
532 OSD
OSD
518 DEMUX
DATA
TD
514 Demodulator
516 PID Filter
IF
512 Tuner
RF
from Forward Channel
TUNE
570 Modulator
to Back Channel
552 Input/Output
558 I/R
580
550
Controller
560 Memory
562 User Interaction Routine
566 Overlay Storage
568 Stream Processing Routine
554 Processor
556 Support Circuits
BS1
BS2
BSN
M
M
M

FIG. 5

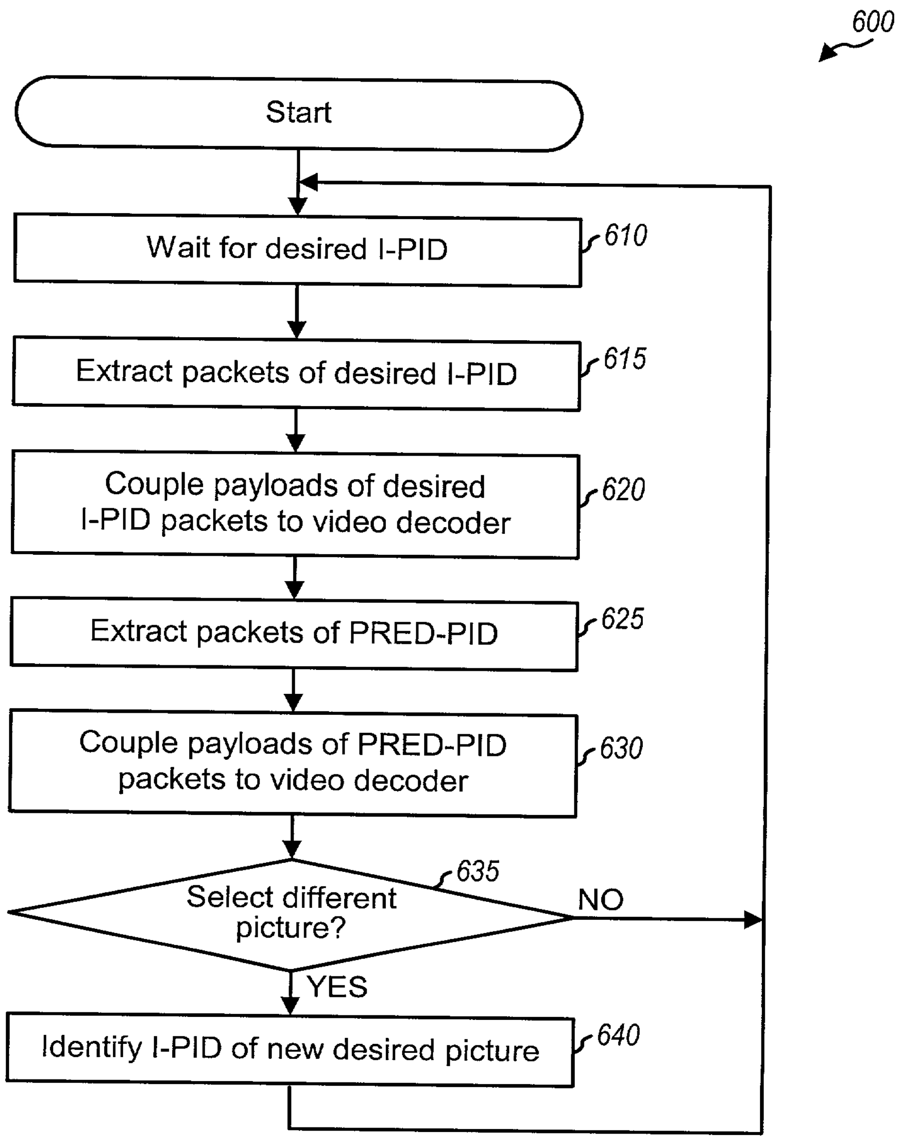

600
Start
Wait for desired I-PID
610
Extract packets of desired I-PID
615
Couple payloads of desired I-PID packets to video decoder
620
Extract packets of PRED-PID
625
Couple payloads of PRED-PID packets to video decoder
630
Select different picture?
635
NO
YES
Identify I-PID of new desired picture
640

FIG. 6

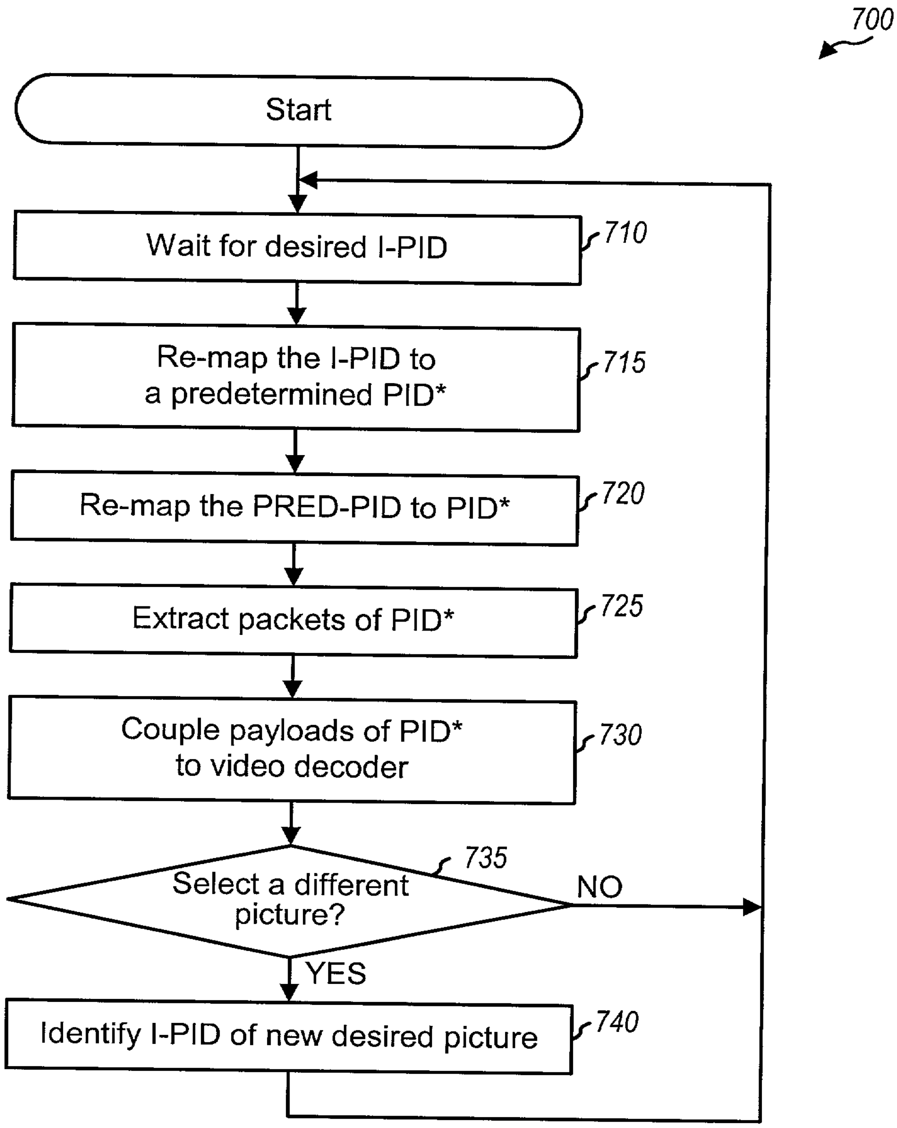

700
Start
Wait for desired I-PID
710
Re-map the I-PID to
a predetermined PID*
715
Re-map the PRED-PID to PID*
720
Extract packets of PID*
725
Couple payloads of PID*
to video decoder
730
Select a different
picture?
735
NO
YES
Identify I-PID of new desired picture
740

FIG. 7

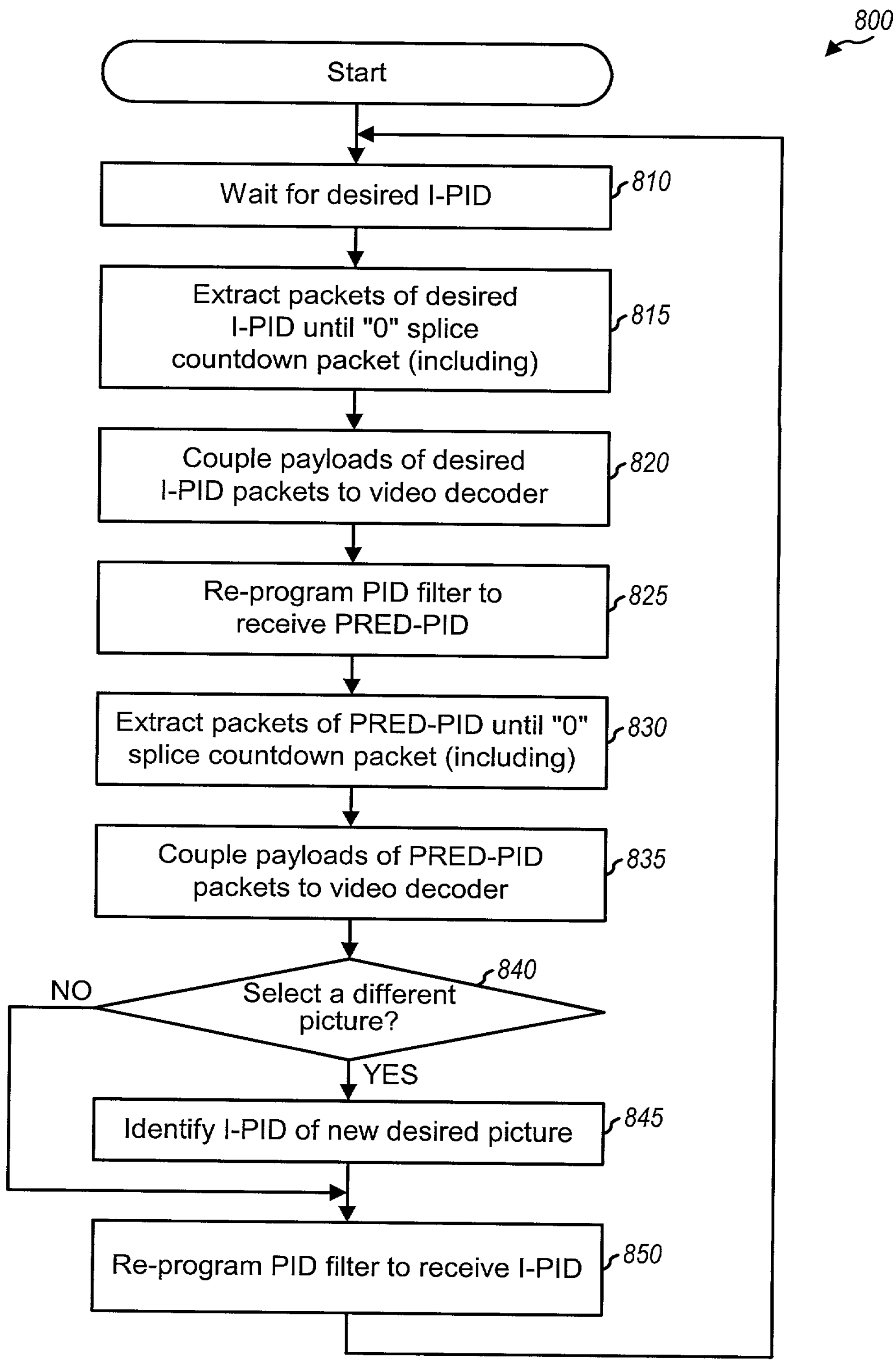

800
Start
Wait for desired I-PID
810
Extract packets of desired I-PID until "0" splice countdown packet (including)
815
Couple payloads of desired I-PID packets to video decoder
820
Re-program PID filter to receive PRED-PID
825
Extract packets of PRED-PID until "0" splice countdown packet (including)
830
Couple payloads of PRED-PID packets to video decoder
835
Select a different picture?
840
NO
YES
Identify I-PID of new desired picture
845
Re-program PID filter to receive I-PID
850

FIG. 8

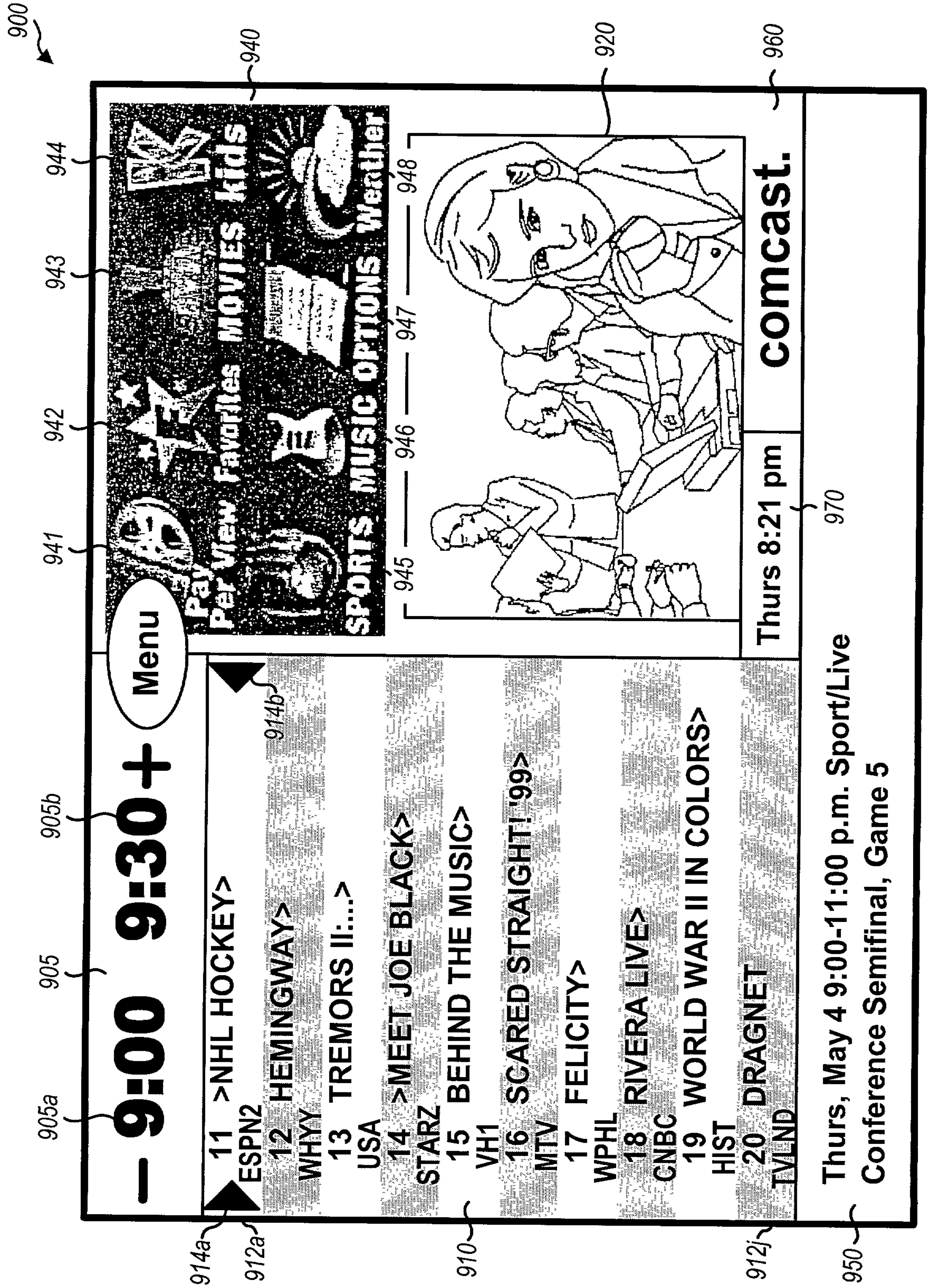

900
940
944
943
942
941
920
960
948
947
946
945
970
905a
905
905b
914b
914a
912a
910
912j
950
– 9:00 9:30 +
Menu
Pay Per View
Favorites
MOVIES
kids
SPORTS
MUSIC
OPTIONS
Weather
comcast.
Thurs 8:21 pm
11 >NHL HOCKEY>
ESPN2
12 HEMINGWAY>
WHYY
13 TREMORS II:...>
USA
14 >MEET JOE BLACK>
STARZ
15 BEHIND THE MUSIC>
VH1
16 SCARED STRAIGHT! '99>
MTV
17 FELICITY>
WPHL
18 RIVERA LIVE>
CNBC
19 WORLD WAR II IN COLORS>
HIST
20 DRAGNET
TVLND
Thurs, May 4 9:00-11:00 p.m. Sport/Live
Conference Semifinal, Game 5

FIG. 9

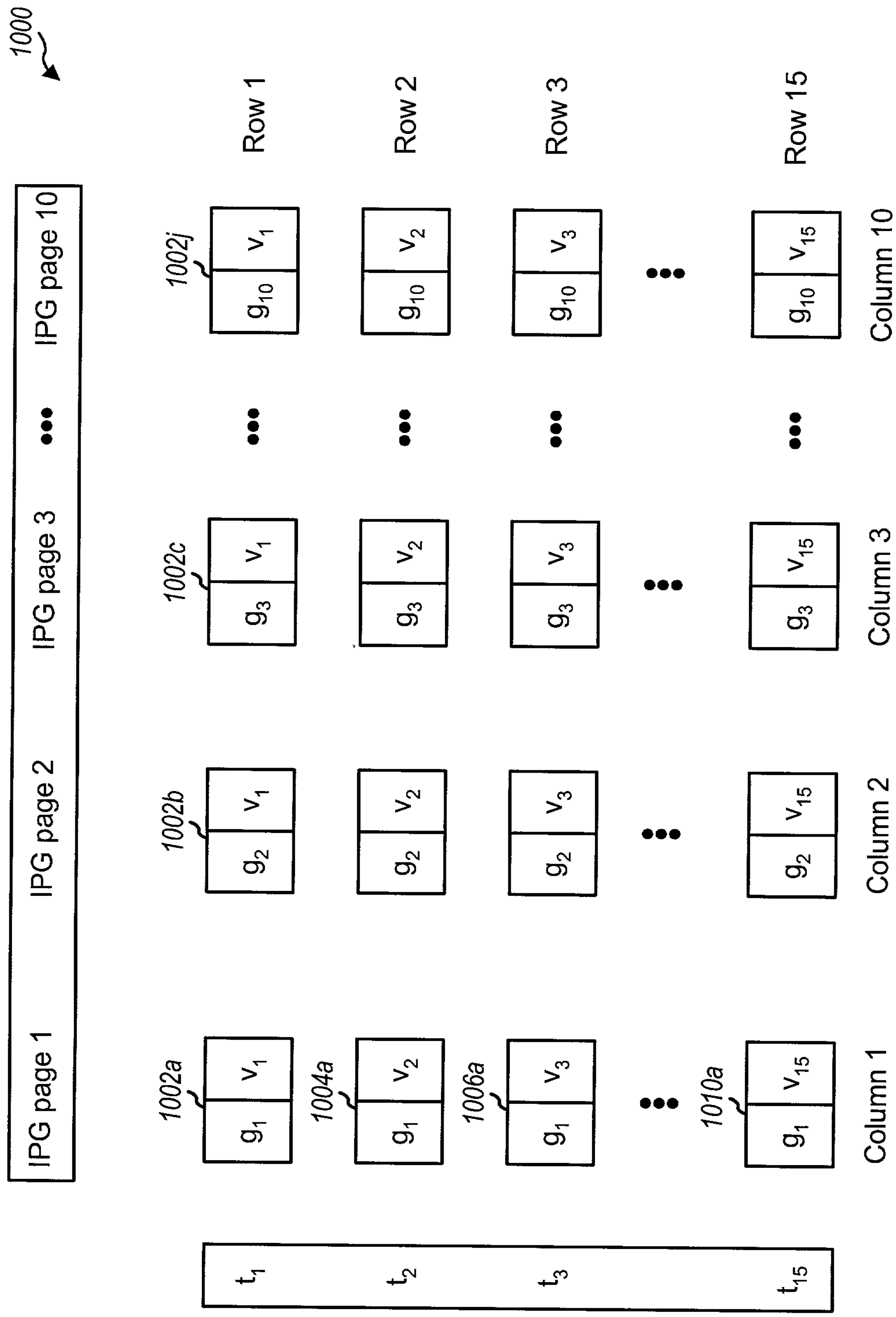

1000
IPG page 1
IPG page 2
IPG page 3
IPG page 10
Row 1
Row 2
Row 3
Row 15
1002a
1004a
1006a
1010a
1002b
1002c
1002j
g1 v1
g1 v2
g1 v3
g1 v15
g2 v1
g2 v2
g2 v3
g2 v15
g3 v1
g3 v2
g3 v3
g3 v15
g10 v1
g10 v2
g10 v3
g10 v15
t1
t2
t3
t15
Column 1
Column 2
Column 3
Column 10

FIG. 10A

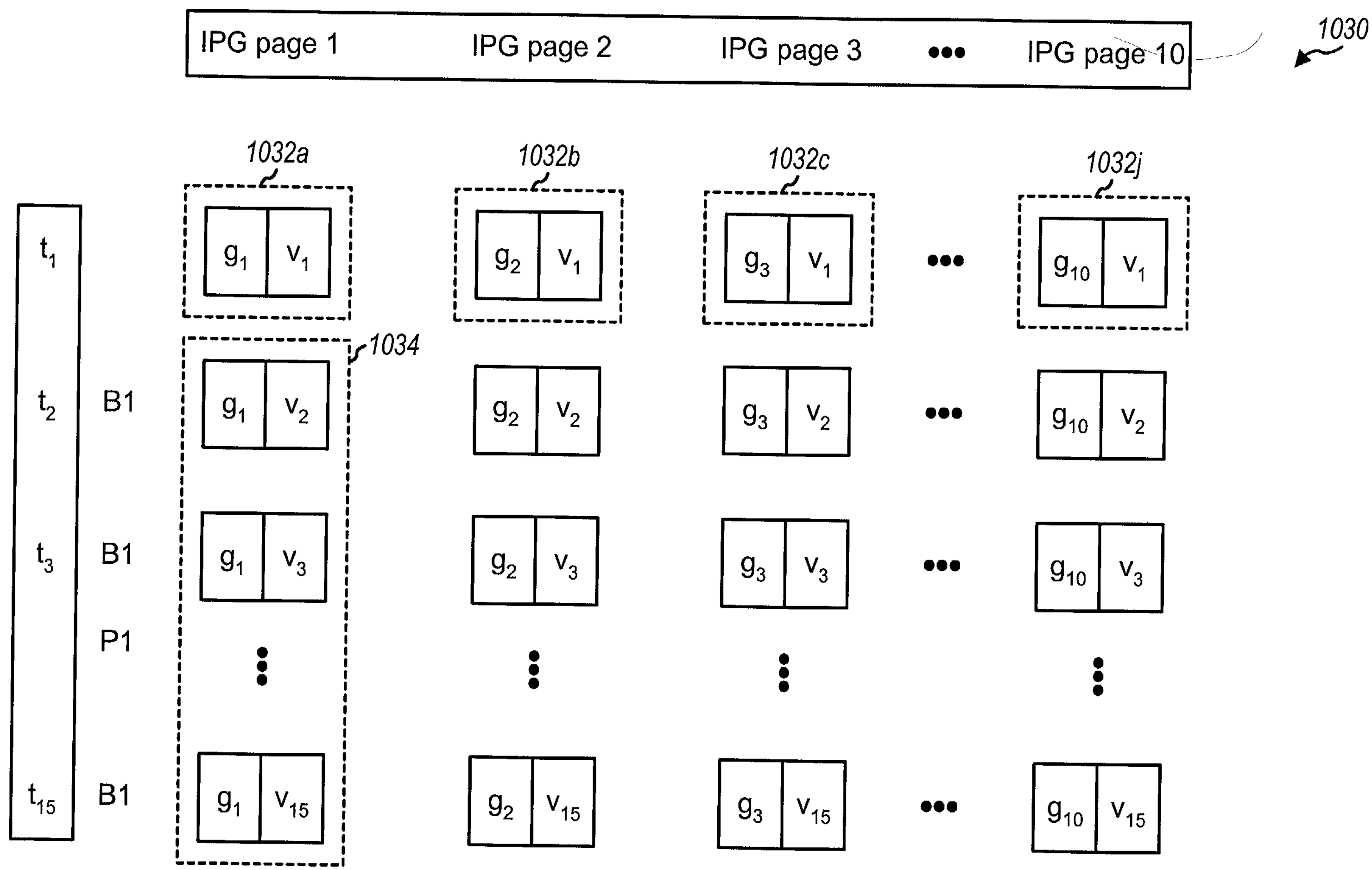

1030
IPG page 1
IPG page 2
IPG page 3
IPG page 10
1032a
1032b
1032c
1032j
1034
t_1
t_2
t_3
t_{15}
B1
B1
P1
B1
g_1 v_1
g_2 v_1
g_3 v_1
g_{10} v_1
g_1 v_2
g_2 v_2
g_3 v_2
g_{10} v_2
g_1 v_3
g_2 v_3
g_3 v_3
g_{10} v_3
g_1 v_{15}
g_2 v_{15}
g_3 v_{15}
g_{10} v_{15}

FIG. 10B

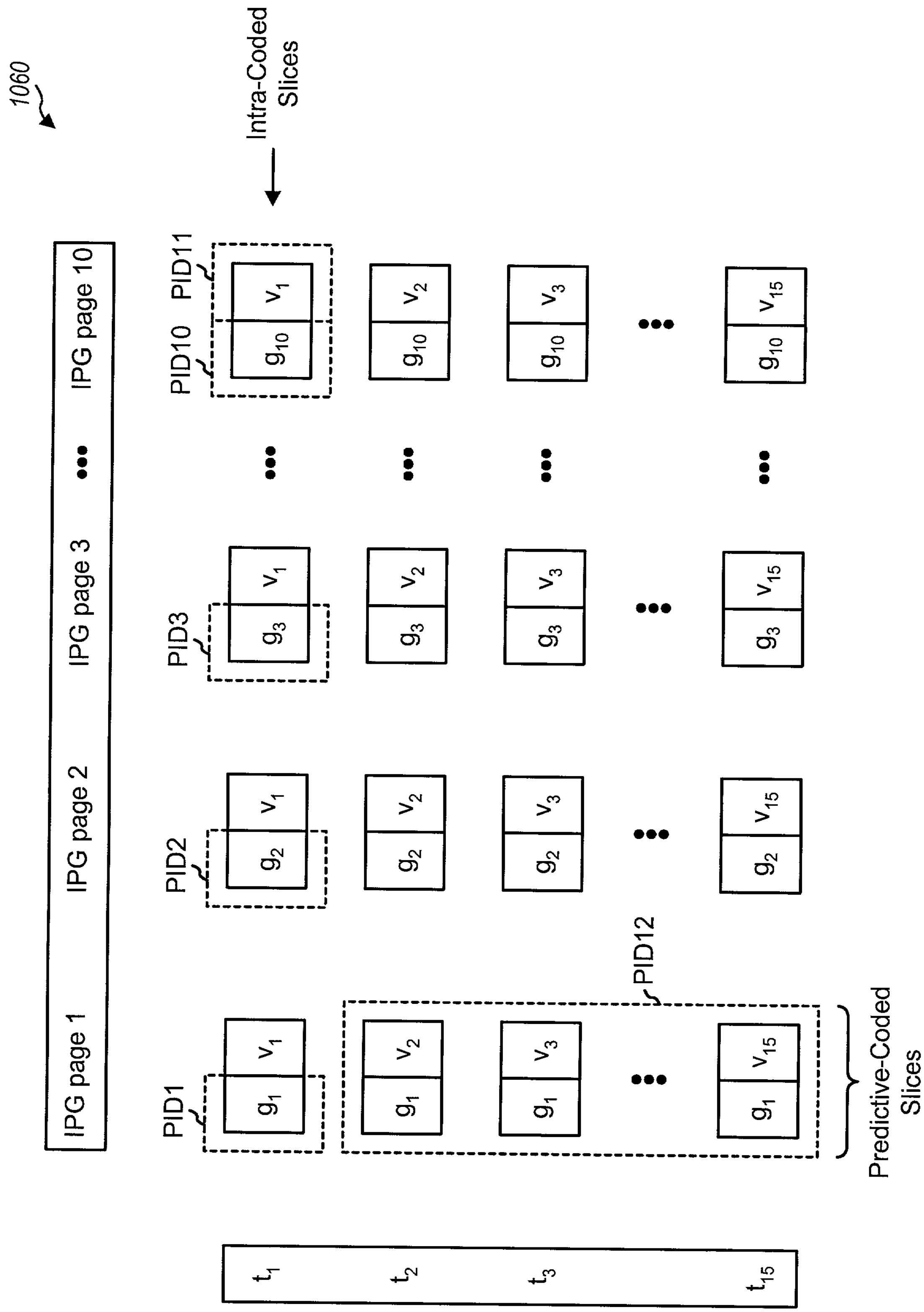

1060
IPG page 1
IPG page 2
IPG page 3
IPG page 10
Intra-Coded Slices
PID1
PID2
PID3
PID10
PID11
PID12
g_1 v_1
g_1 v_2
g_1 v_3
g_1 v_{15}
g_2 v_1
g_2 v_2
g_2 v_3
g_2 v_{15}
g_3 v_1
g_3 v_2
g_3 v_3
g_3 v_{15}
g_{10} v_1
g_{10} v_2
g_{10} v_3
g_{10} v_{15}
Predictive-Coded Slices
t_1
t_2
t_3
t_{15}

FIG. 10C

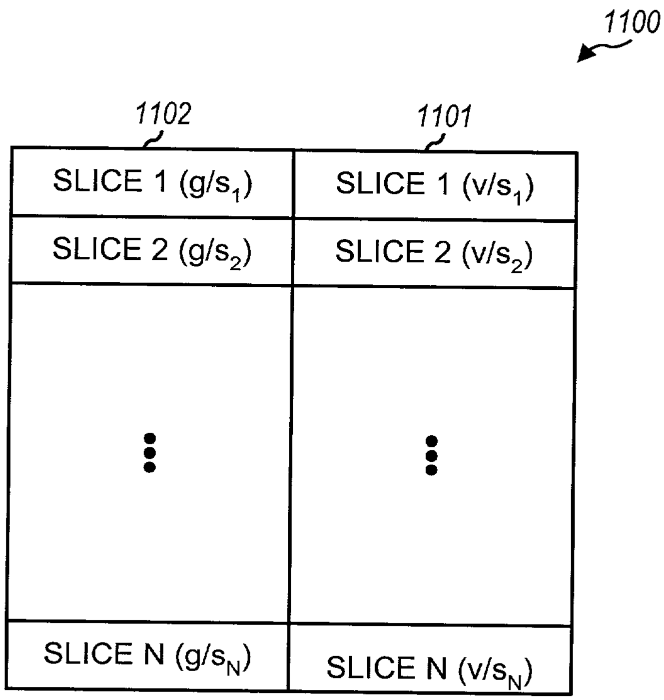

1100
1102
1101
SLICE 1 (g/s1)
SLICE 1 (v/s1)
SLICE 2 (g/s2)
SLICE 2 (v/s2)
SLICE N (g/sN)
SLICE N (v/sN)

FIG. 11

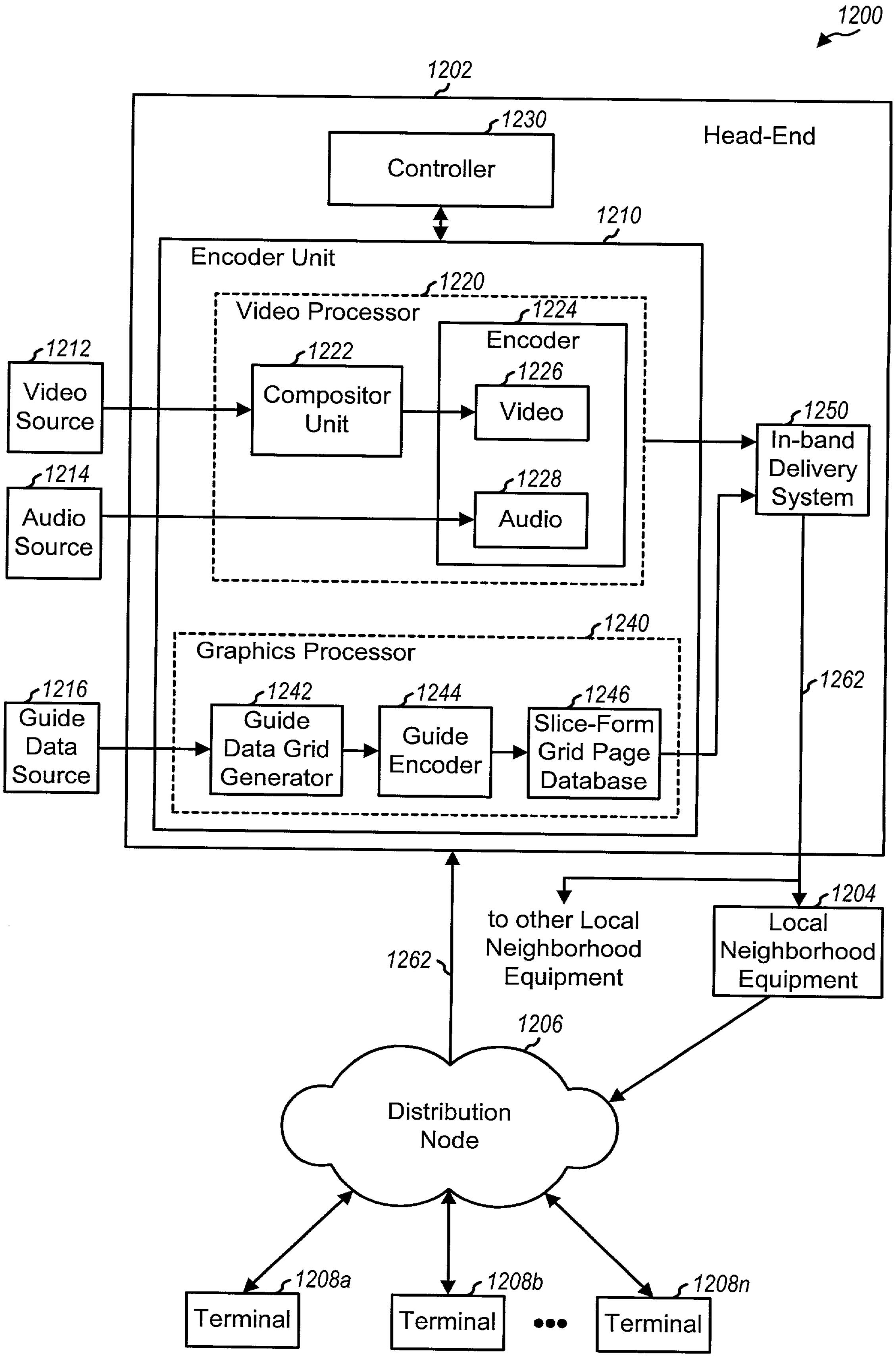

1200
1202
Head-End
1230
Controller
1210
Encoder Unit
1220
Video Processor
1224
Encoder
1212
Video Source
1222
Compositor Unit
1226
Video
1250
In-band Delivery System
1214
Audio Source
1228
Audio
1240
Graphics Processor
1262
1216
Guide Data Source
1242
Guide Data Grid Generator
1244
Guide Encoder
1246
Slice-Form Grid Page Database
1204
to other Local Neighborhood Equipment
Local Neighborhood Equipment
1262
1206
Distribution Node
1208a
Terminal
1208b
Terminal
1208n
Terminal

FIG. 12A

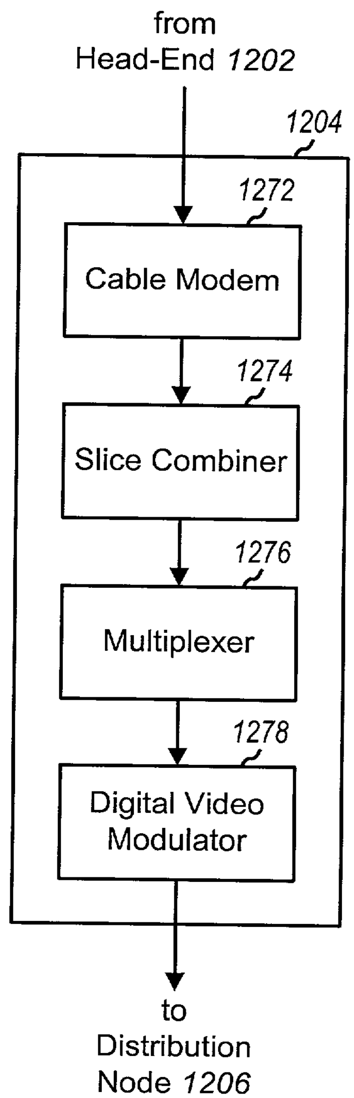

from
Head-End 1202
1204
1272
Cable Modem
1274
Slice Combiner
1276
Multiplexer
1278
Digital Video
Modulator
to
Distribution
Node 1206

FIG. 12B

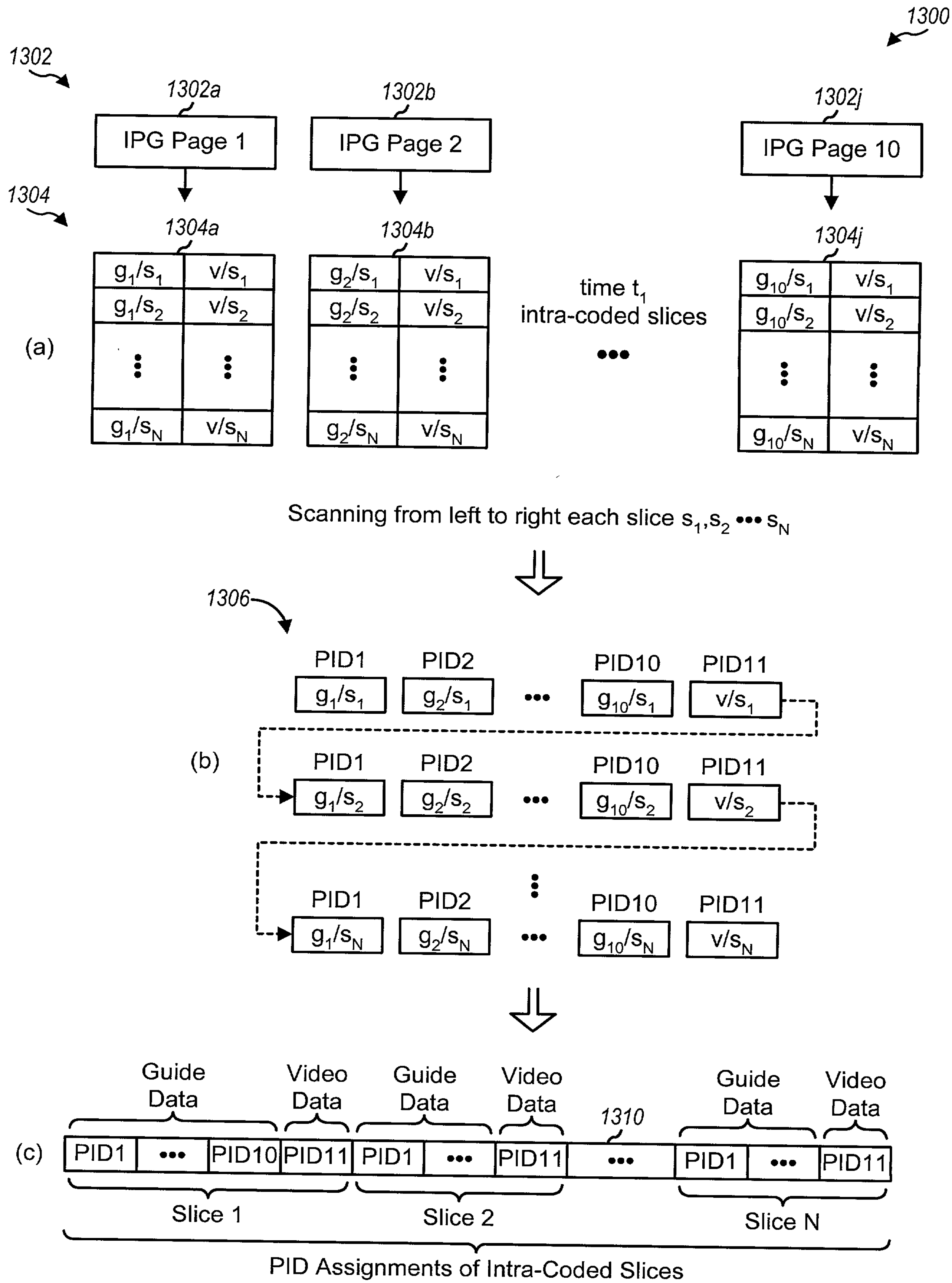

1300
1302
1302a
IPG Page 1
1302b
IPG Page 2
1302j
IPG Page 10
1304
1304a
1304b
1304j
(a)
g1/s1 v/s1
g1/s2 v/s2
g1/sN v/sN
g2/s1 v/s1
g2/s2 v/s2
g2/sN v/sN
time t1
intra-coded slices
g10/s1 v/s1
g10/s2 v/s2
g10/sN v/sN
Scanning from left to right each slice s1,s2 ••• sN
1306
(b)
PID1 PID2 PID10 PID11
g1/s1 g2/s1 g10/s1 v/s1
g1/s2 g2/s2 g10/s2 v/s2
g1/sN g2/sN g10/sN v/sN
(c)
Guide Data
Video Data
1310
PID1 ••• PID10 PID11 PID1 ••• PID11 ••• PID1 ••• PID11
Slice 1
Slice 2
Slice N
PID Assignments of Intra-Coded Slices

FIG. 13

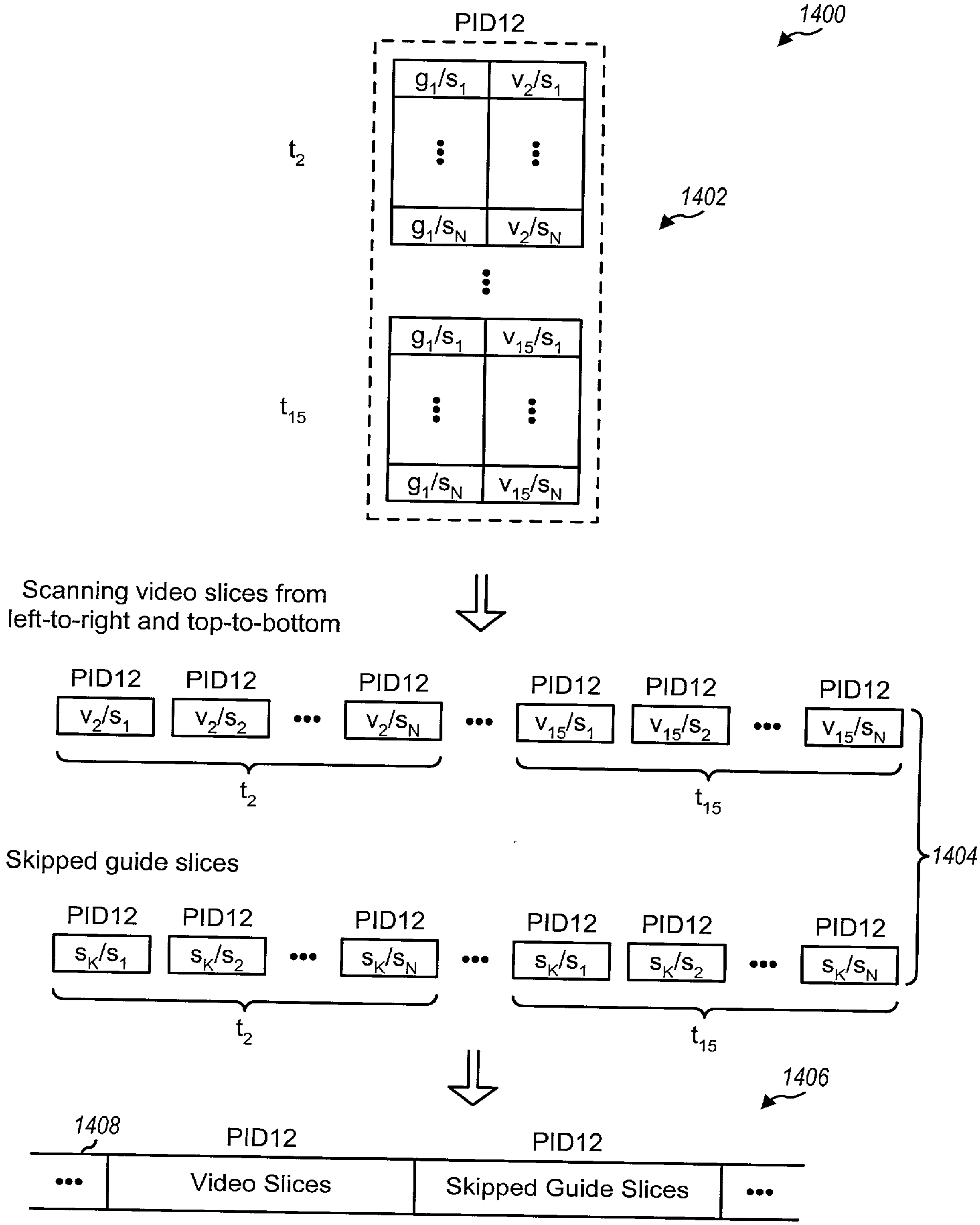

1400
PID12
g_1/s_1
v_2/s_1
t_2
1402
g_1/s_N
v_2/s_N
g_1/s_1
v_{15}/s_1
t_{15}
g_1/s_N
v_{15}/s_N
Scanning video slices from left-to-right and top-to-bottom
PID12
v_2/s_1
v_2/s_2
v_2/s_N
v_{15}/s_1
v_{15}/s_2
v_{15}/s_N
t_2
t_{15}
1404
Skipped guide slices
s_K/s_1
s_K/s_2
s_K/s_N
1406
1408
Video Slices
Skipped Guide Slices

FIG. 14

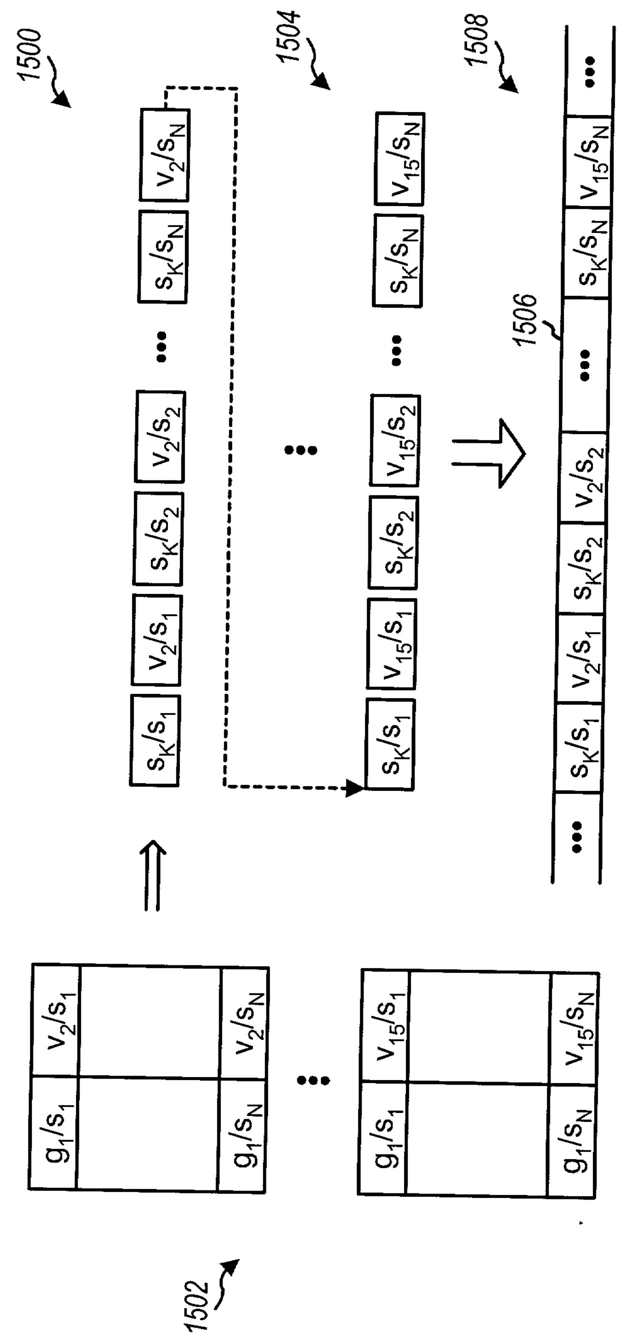

1500
1502
1504
1506
1508
g_1/s_1
v_2/s_1
g_1/s_N
v_2/s_N
v_{15}/s_1
v_{15}/s_N
s_K/s_1
s_K/s_2
v_2/s_2
s_K/s_N
v_{15}/s_2

FIG. 15

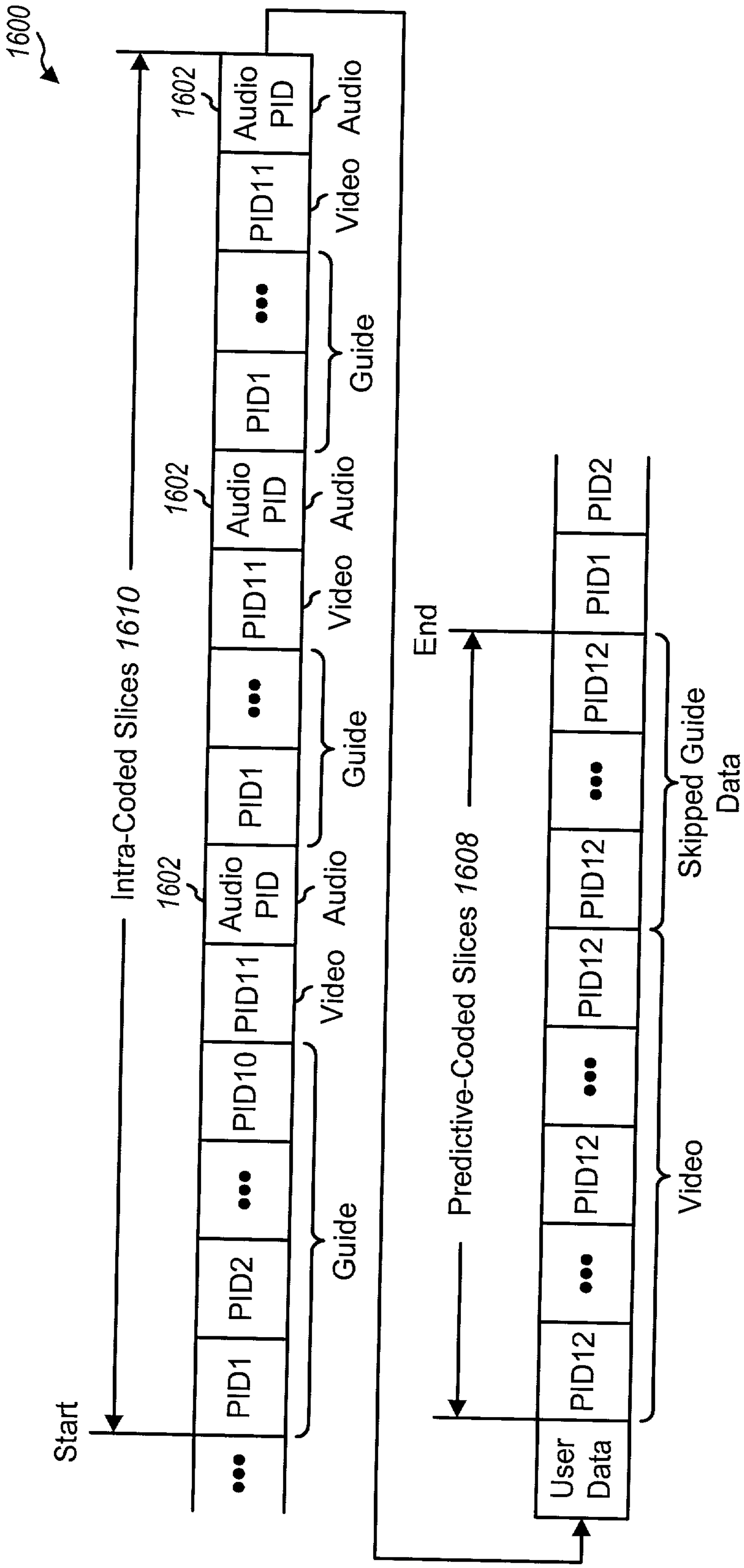

1600
Intra-Coded Slices 1610
Start
PID1
PID2
•••
PID10
PID11
Audio PID
1602
Guide
Video
Audio
PID1
•••
PID11
Audio PID
1602
Guide
Video
Audio
PID1
•••
PID11
Audio PID
1602
Guide
Video
Audio
Predictive-Coded Slices 1608
End
User Data
PID12
•••
PID12
•••
PID12
PID12
•••
PID12
PID1
PID2
Video
Skipped Guide Data

FIG. 16

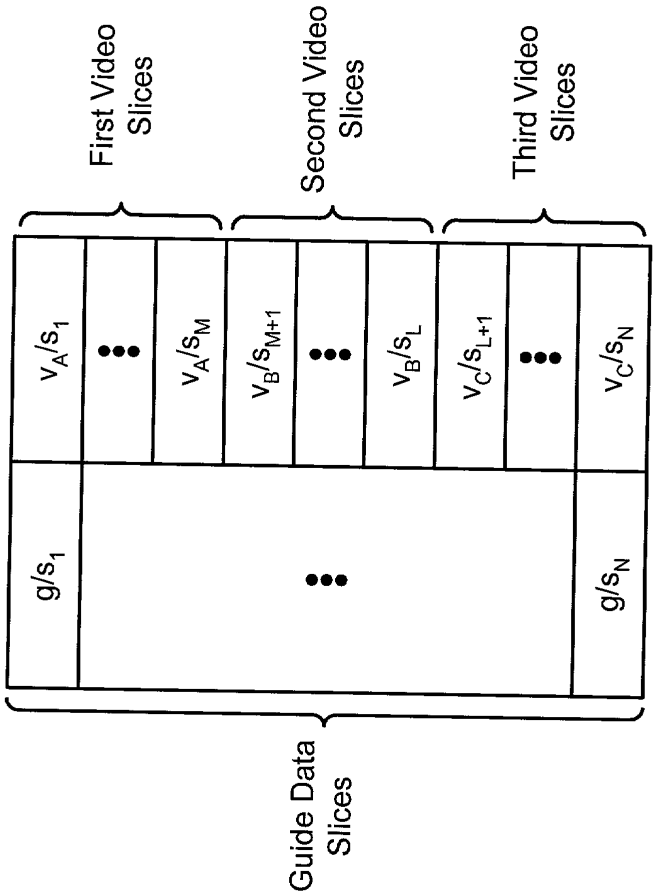


FIG. 17B

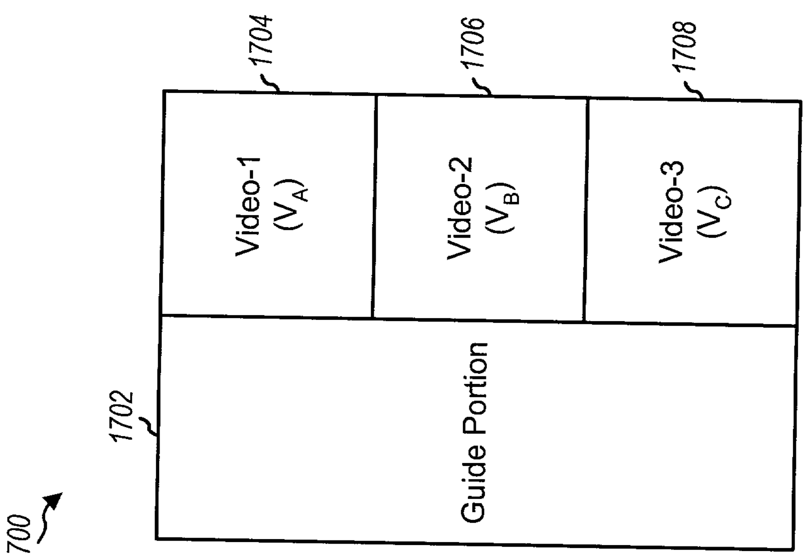


FIG. 17A

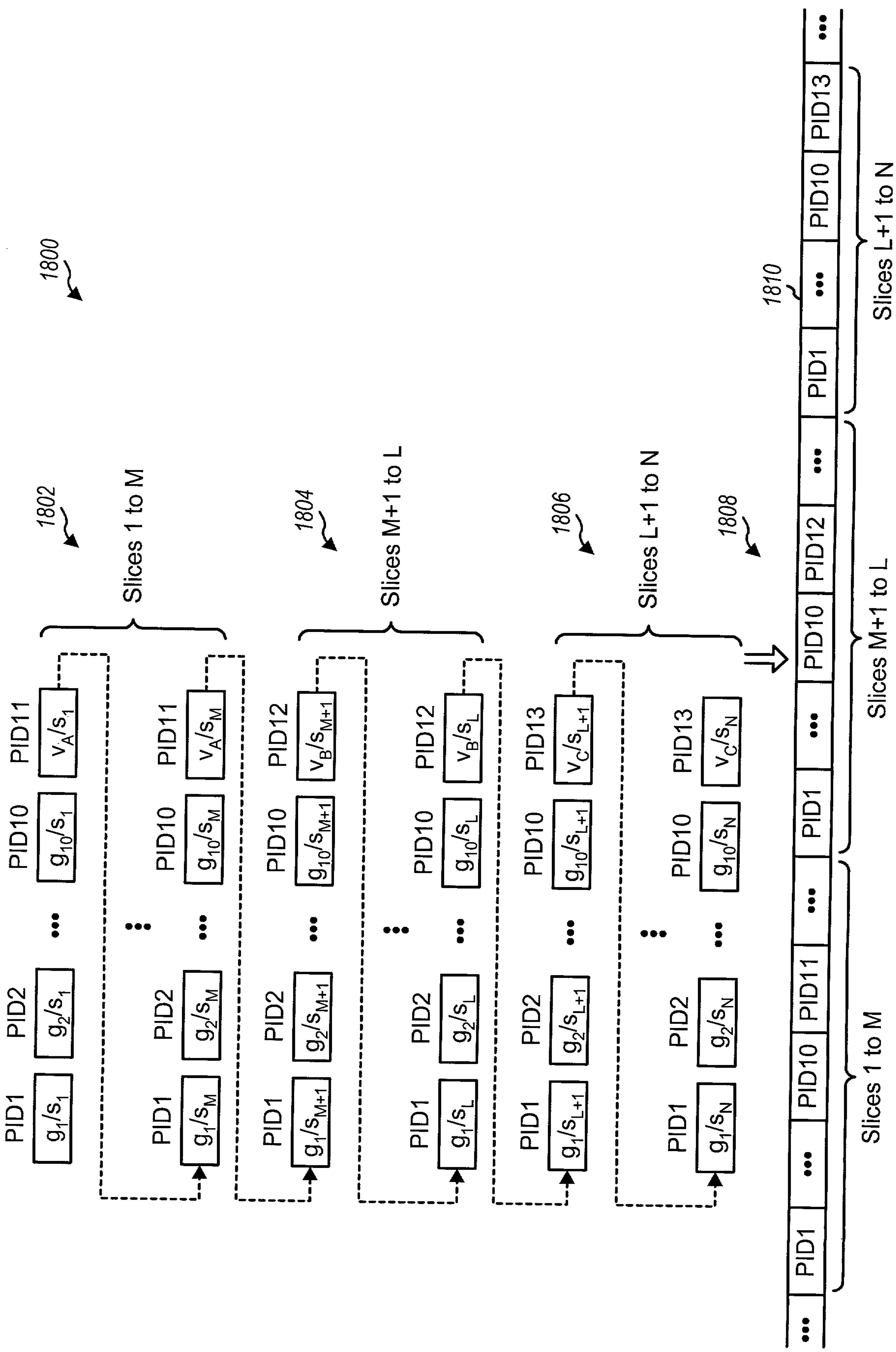

1800
1802
1804
1806
1808
1810
Slices 1 to M
Slices M+1 to L
Slices L+1 to N
PID1 g_1/s_1
PID2 g_2/s_1
PID10 g_{10}/s_1
PID11 v_A/s_1
PID1 g_1/s_M
PID2 g_2/s_M
PID10 g_{10}/s_M
PID11 v_A/s_M
PID1 g_1/s_{M+1}
PID2 g_2/s_{M+1}
PID10 g_{10}/s_{M+1}
PID12 v_B/s_{M+1}
PID1 g_1/s_L
PID2 g_2/s_L
PID10 g_{10}/s_L
PID12 v_B/s_L
PID1 g_1/s_{L+1}
PID2 g_2/s_{L+1}
PID10 g_{10}/s_{L+1}
PID13 v_C/s_{L+1}
PID1 g_1/s_N
PID2 g_2/s_N
PID10 g_{10}/s_N
PID13 v_C/s_N
PID1 PID10 PID11 PID1 PID10 PID12 PID1 PID10 PID13
Slices 1 to M
Slices M+1 to L
Slices L+1 to N

FIG. 18

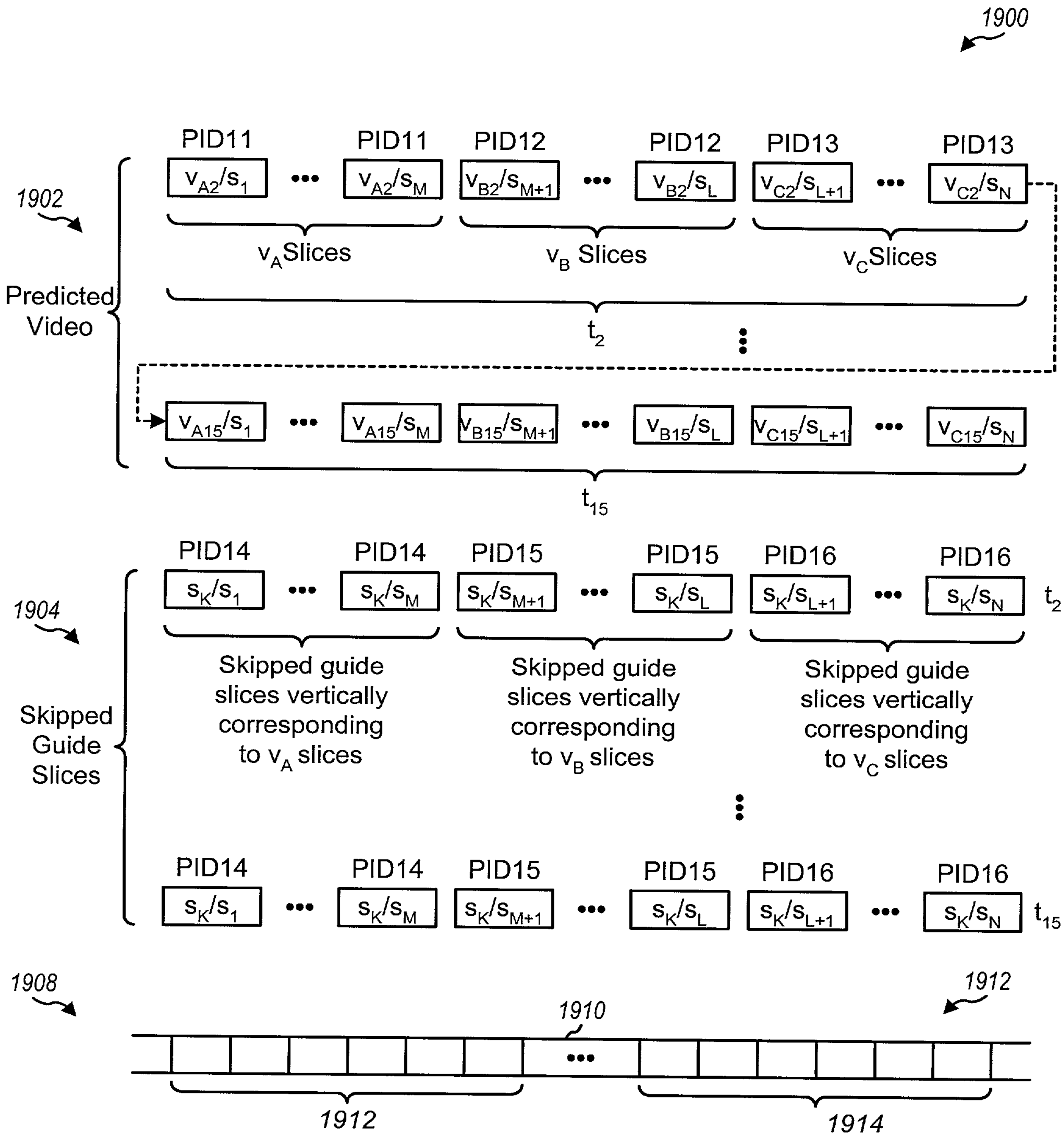

1900
1902
Predicted Video
PID11 PID11 PID12 PID12 PID13 PID13
v_{A2}/s_1 ••• v_{A2}/s_M v_{B2}/s_{M+1} ••• v_{B2}/s_L v_{C2}/s_{L+1} ••• v_{C2}/s_N
v_A Slices
v_B Slices
v_C Slices
t_2
v_{A15}/s_1 ••• v_{A15}/s_M v_{B15}/s_{M+1} ••• v_{B15}/s_L v_{C15}/s_{L+1} ••• v_{C15}/s_N
t_{15}
1904
Skipped Guide Slices
PID14 PID14 PID15 PID15 PID16 PID16
s_K/s_1 ••• s_K/s_M s_K/s_{M+1} ••• s_K/s_L s_K/s_{L+1} ••• s_K/s_N t_2
Skipped guide slices vertically corresponding to v_A slices
Skipped guide slices vertically corresponding to v_B slices
Skipped guide slices vertically corresponding to v_C slices
PID14 PID14 PID15 PID15 PID16 PID16
s_K/s_1 ••• s_K/s_M s_K/s_{M+1} ••• s_K/s_L s_K/s_{L+1} ••• s_K/s_N t_{15}
1908
1910
1912
1912
1914

FIG. 19

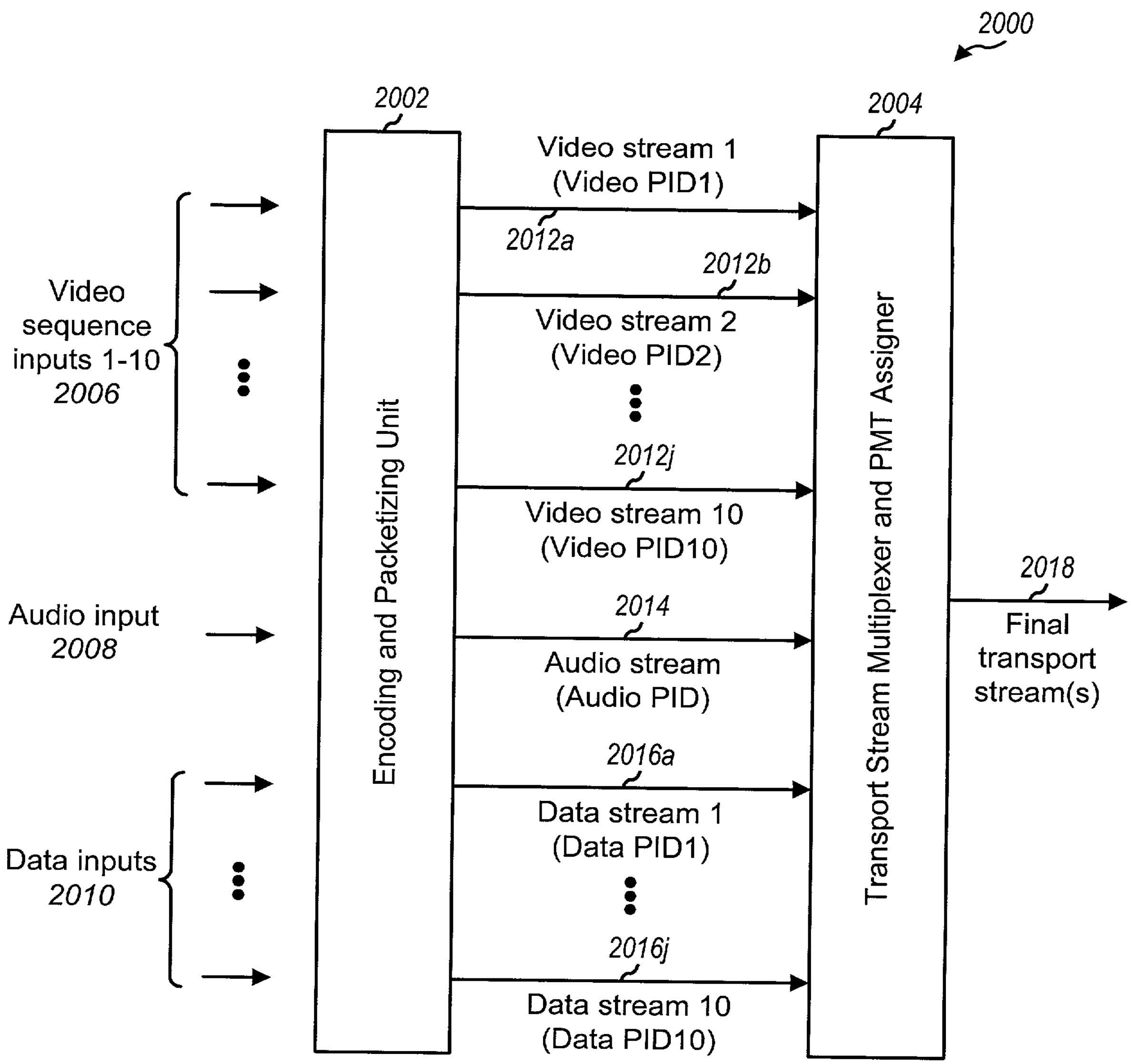
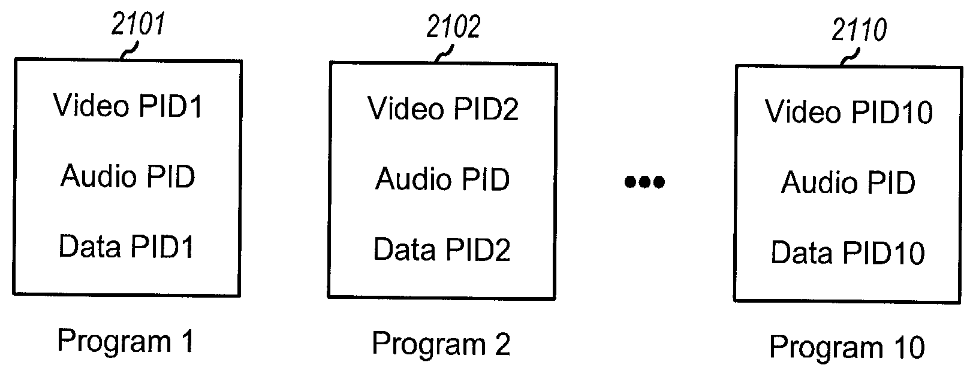
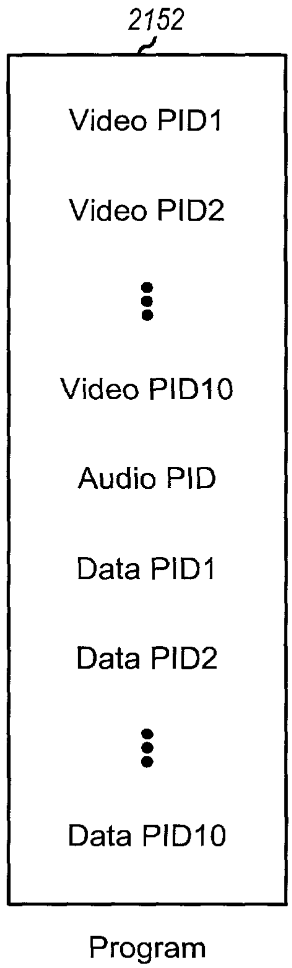

2000
2002
2004
Video sequence inputs 1-10
2006
Audio input
2008
Data inputs
2010
Encoding and Packetizing Unit
Video stream 1
(Video PID1)
2012a
2012b
Video stream 2
(Video PID2)
2012j
Video stream 10
(Video PID10)
2014
Audio stream
(Audio PID)
2016a
Data stream 1
(Data PID1)
2016j
Data stream 10
(Data PID10)
Transport Stream Multiplexer and PMT Assigner
2018
Final transport stream(s)

FIG. 20



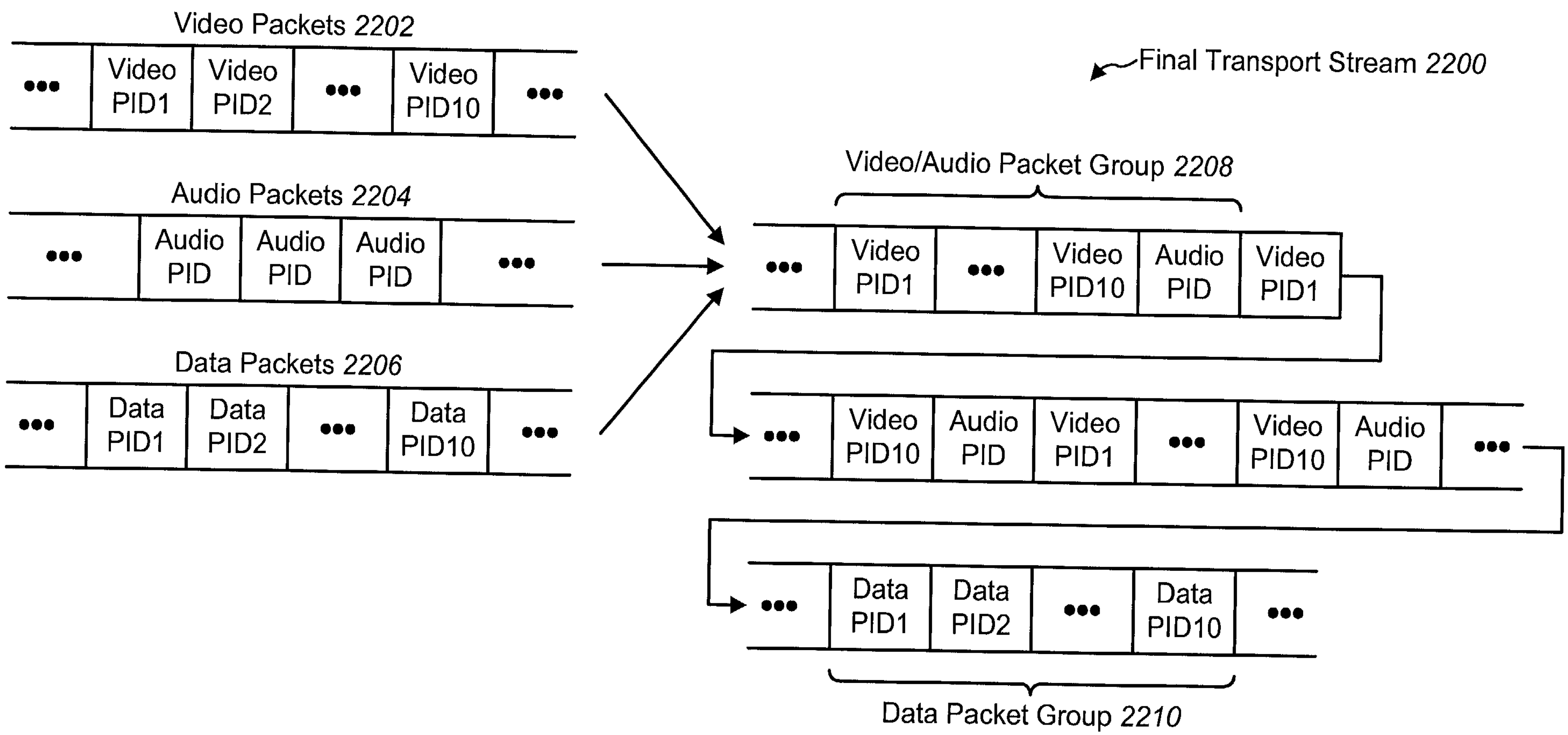
Single Transport Stream, Multiple Programs, Program Assignment 2100

FIG. 21A



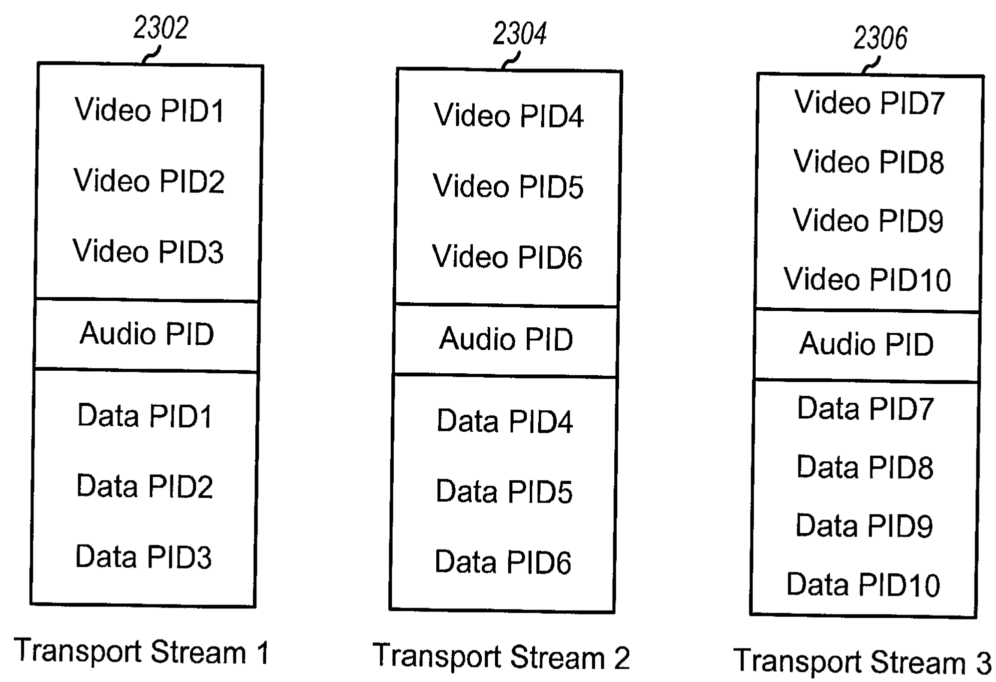
Single Transport Steam, Single Program, Program Assignment 2150

FIG. 21B



Multiplexing Into Single Transport Stream

FIG. 22



Multiple Transport Streams Assignment Structure 2300

FIG. 23

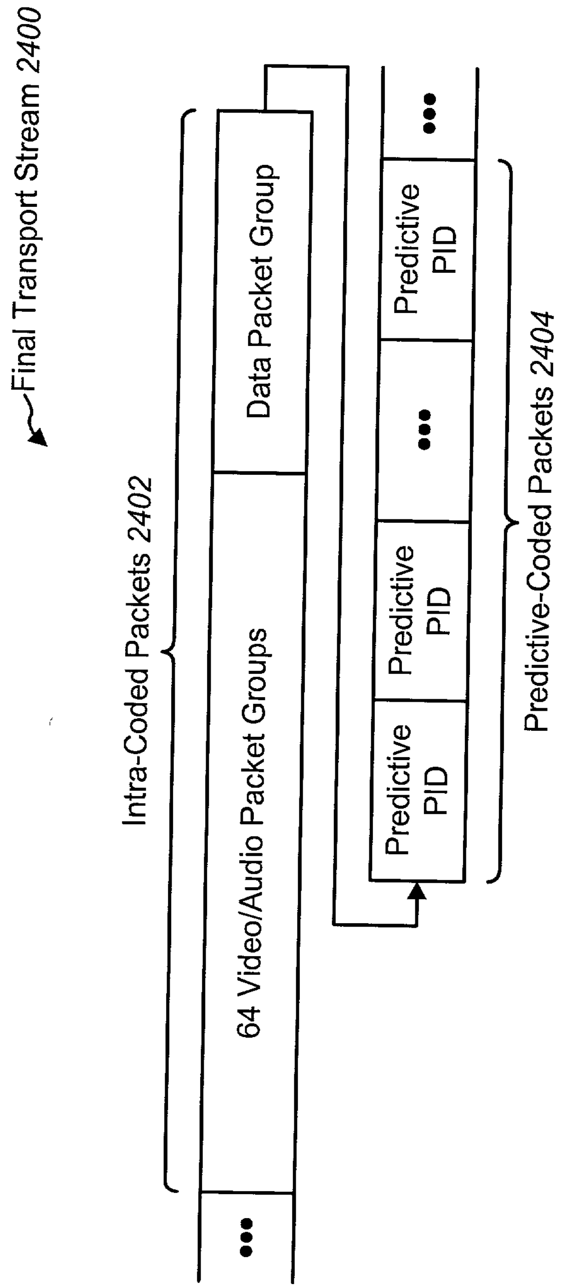

Final Transport Stream 2400
Intra-Coded Packets 2402
64 Video/Audio Packet Groups
Data Packet Group
Predictive PID
Predictive PID
Predictive PID
Predictive-Coded Packets 2404

FIG. 24

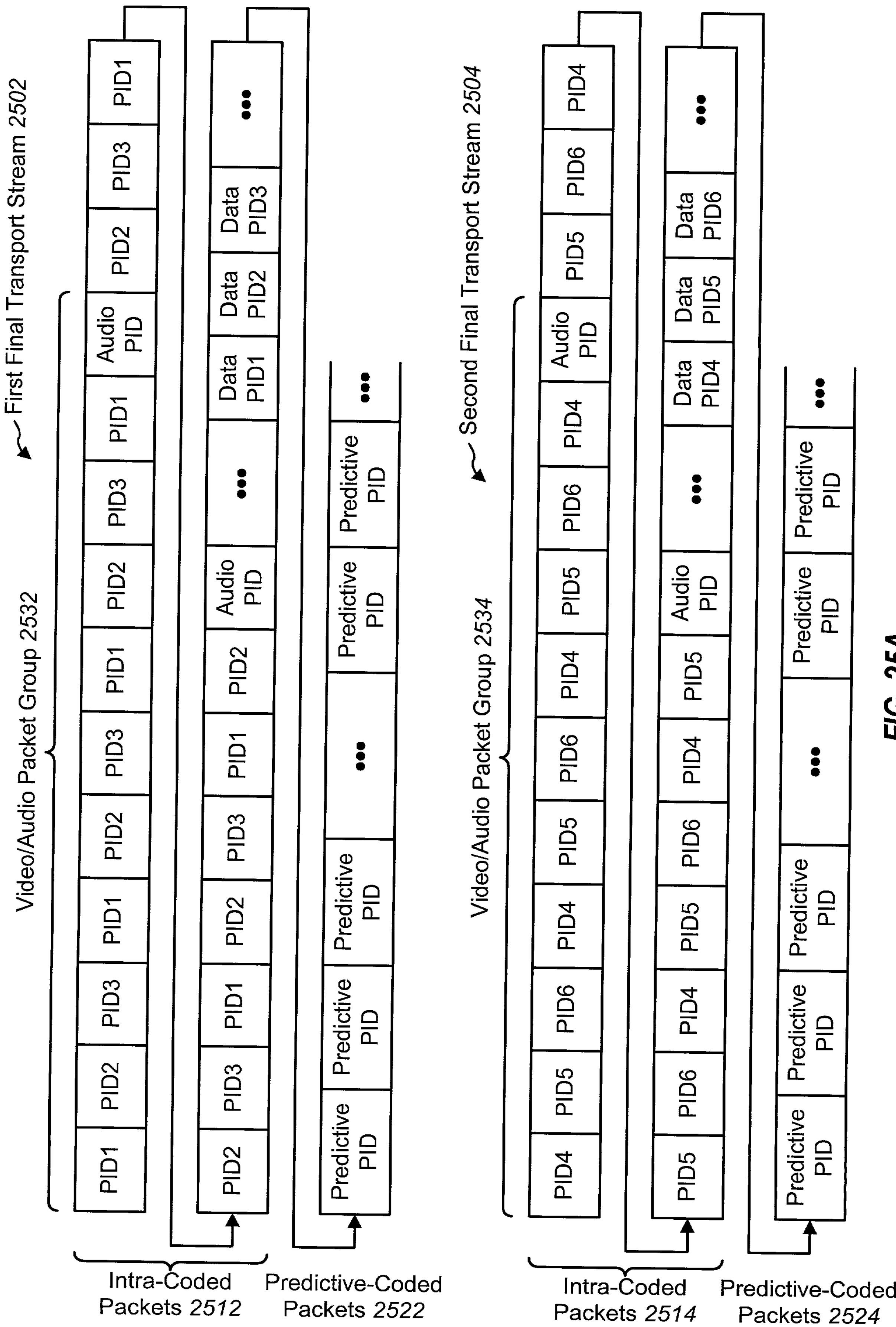

First Final Transport Stream 2502
Video/Audio Packet Group 2532
PID1
PID2
PID3
Audio PID
Data PID1
Data PID2
Data PID3
Predictive PID
Intra-Coded Packets 2512
Predictive-Coded Packets 2522
Second Final Transport Stream 2504
Video/Audio Packet Group 2534
PID4
PID5
PID6
Data PID4
Data PID5
Data PID6
Intra-Coded Packets 2514
Predictive-Coded Packets 2524

FIG. 25A

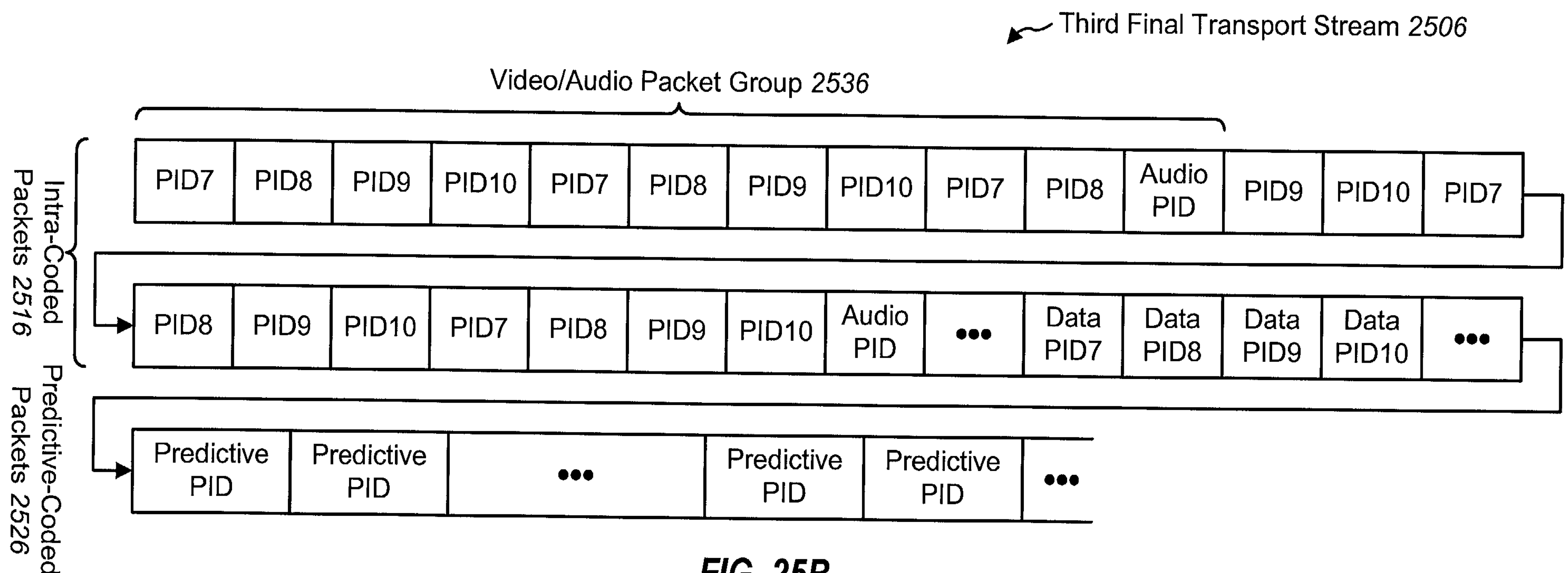

Third Final Transport Stream 2506
Video/Audio Packet Group 2536
PID7
PID8
PID9
PID10
Audio PID
Data PID7
Data PID8
Data PID9
Data PID10
Predictive PID
Intra-Coded Packets 2516
Predictive-Coded Packets 2526

FIG. 25B

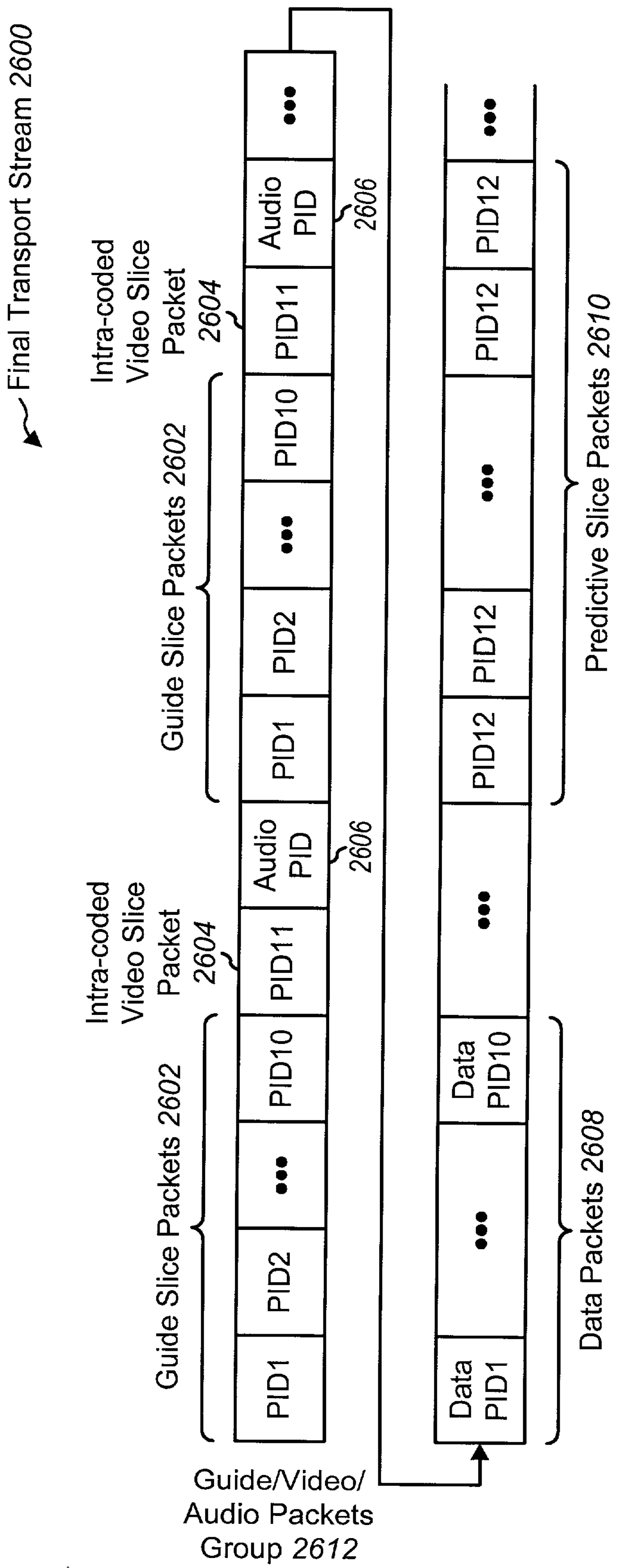

Final Transport Stream 2600
Intra-coded Video Slice Packet 2604
Guide Slice Packets 2602
PID1
PID2
PID10
PID11
Audio PID
2606
Intra-coded Video Slice Packet 2604
Guide Slice Packets 2602
PID1
PID2
PID10
PID11
Audio PID
2606
Data PID1
Data PID10
PID12
PID12
PID12
PID12
Data Packets 2608
Predictive Slice Packets 2610
Guide/Video/ Audio Packets Group 2612

FIG. 26

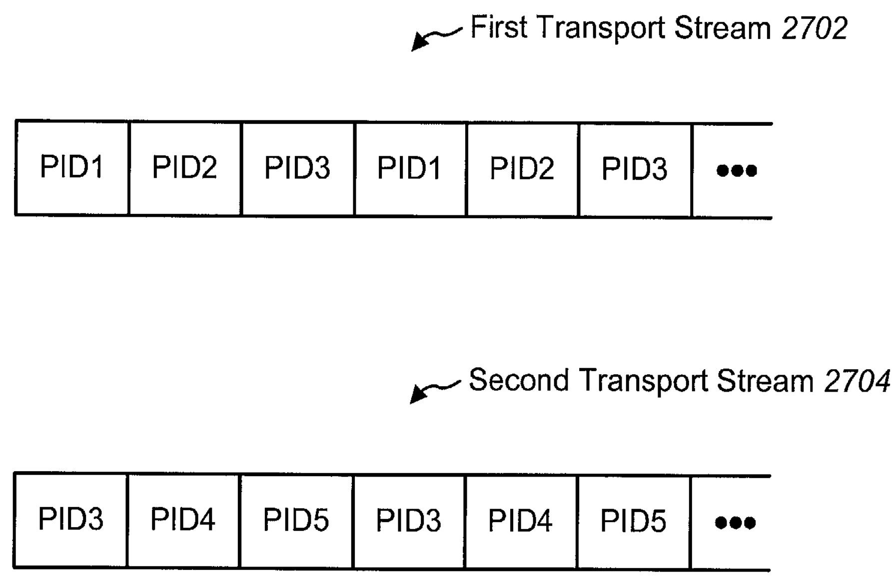

First Transport Stream 2702
PID1
PID2
PID3
PID1
PID2
PID3
Second Transport Stream 2704
PID3
PID4
PID5
PID3
PID4
PID5

FIG. 27A

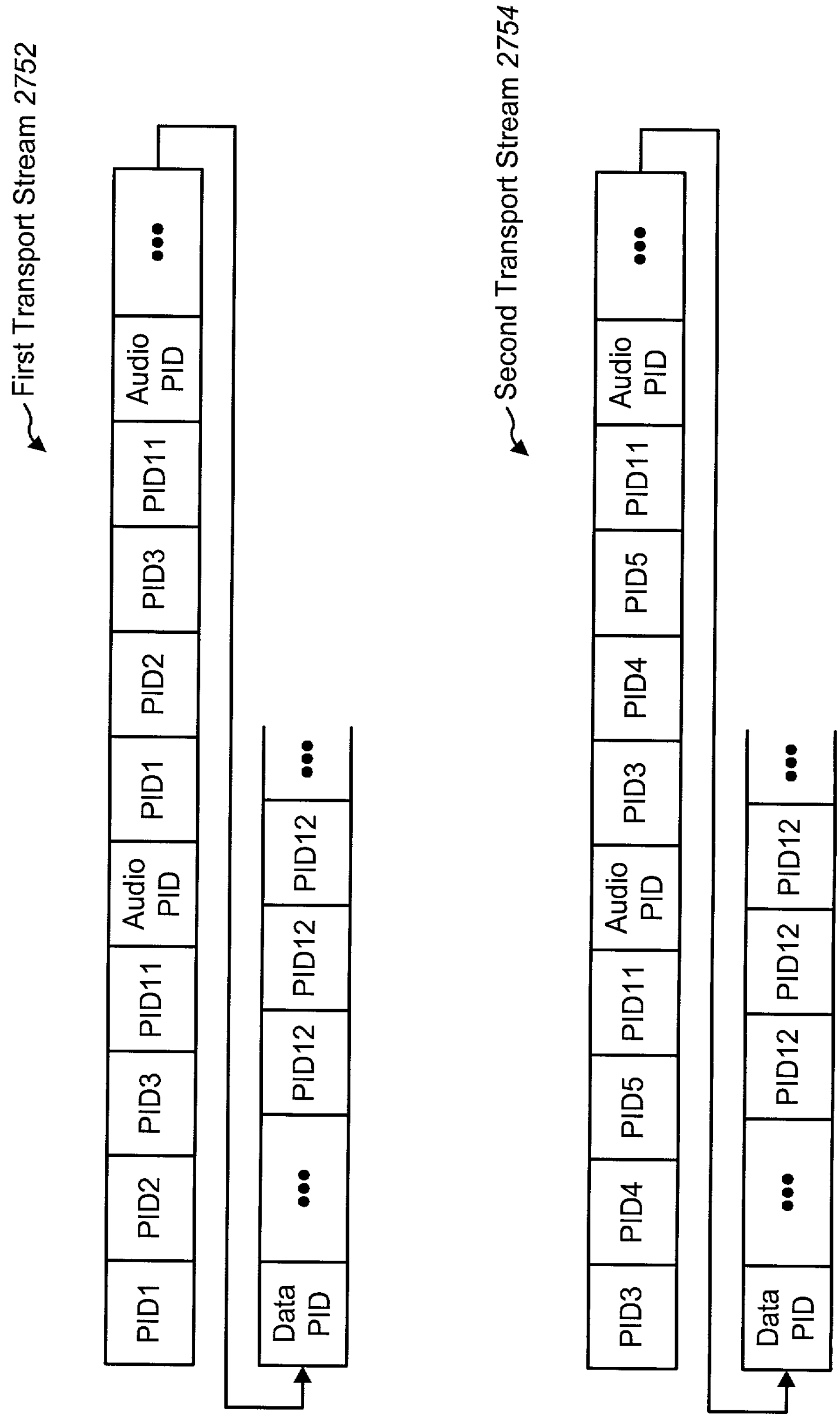

First Transport Stream 2752
PID1
PID2
PID3
PID11
Audio PID
PID1
PID2
PID3
PID11
Audio PID
Data PID
PID12
PID12
PID12
Second Transport Stream 2754
PID3
PID4
PID5
PID11
Audio PID
PID3
PID4
PID5
PID11
Audio PID
Data PID
PID12
PID12
PID12

FIG. 27B

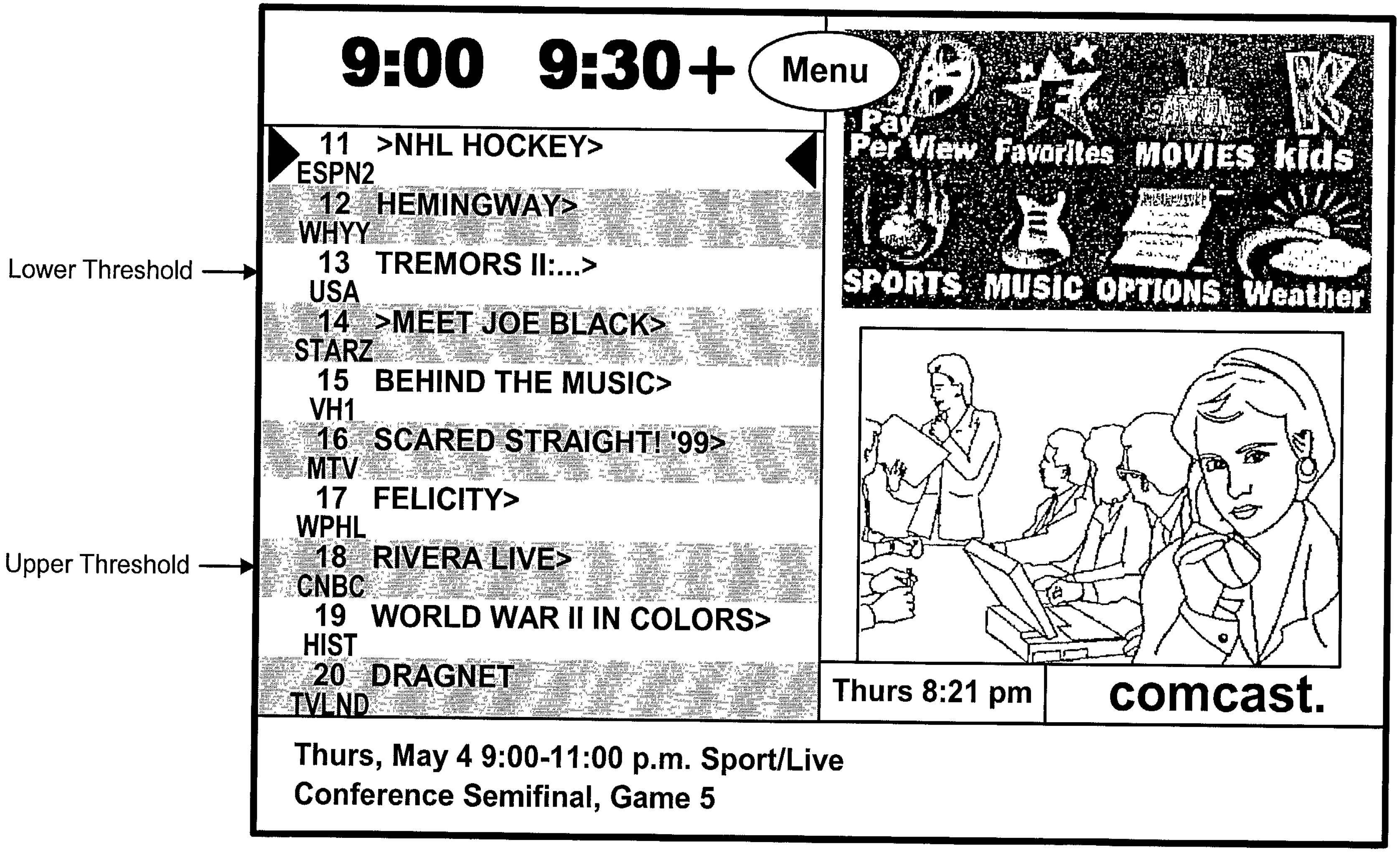

9:00 9:30+
Menu
Pay Per View
Favorites
MOVIES
kids
SPORTS
MUSIC
OPTIONS
Weather
11 >NHL HOCKEY>
ESPN2
12 HEMINGWAY>
WHYY
13 TREMORS II:...>
USA
14 >MEET JOE BLACK>
STARZ
15 BEHIND THE MUSIC>
VH1
16 SCARED STRAIGHT! '99>
MTV
17 FELICITY>
WPHL
18 RIVERA LIVE>
CNBC
19 WORLD WAR II IN COLORS>
HIST
20 DRAGNET
TVLND
Lower Threshold
Upper Threshold
Thurs 8:21 pm
comcast.
Thurs, May 4 9:00-11:00 p.m. Sport/Live
Conference Semifinal, Game 5

FIG. 28

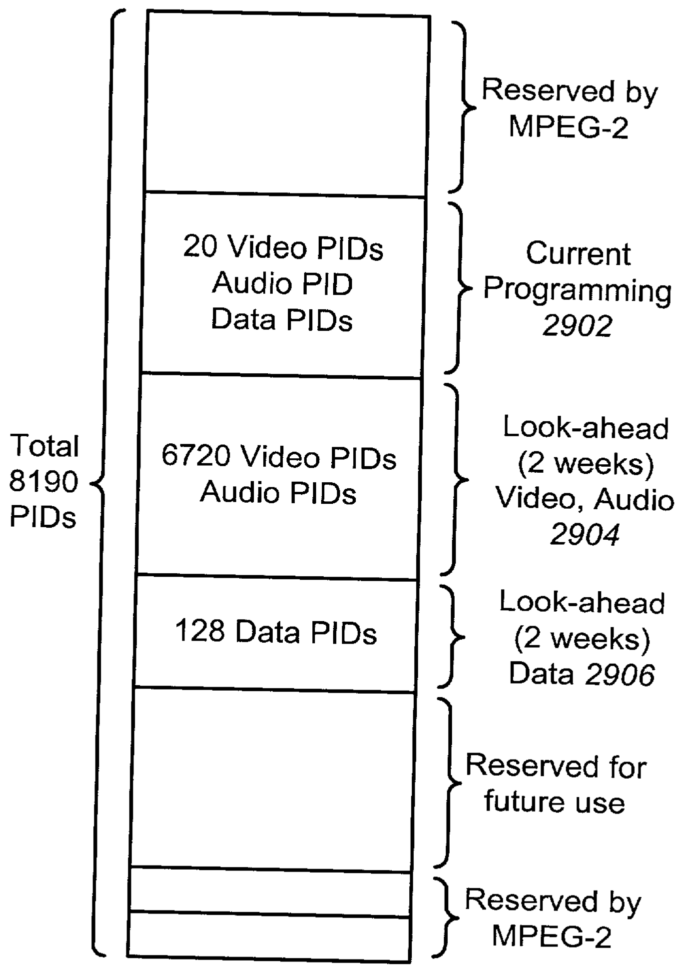

Reserved by
MPEG-2
20 Video PIDs
Audio PID
Data PIDs
Current
Programming
2902
Total
8190
PIDs
6720 Video PIDs
Audio PIDs
Look-ahead
(2 weeks)
Video, Audio
2904
128 Data PIDs
Look-ahead
(2 weeks)
Data 2906
Reserved for
future use
Reserved by
MPEG-2

FIG. 29

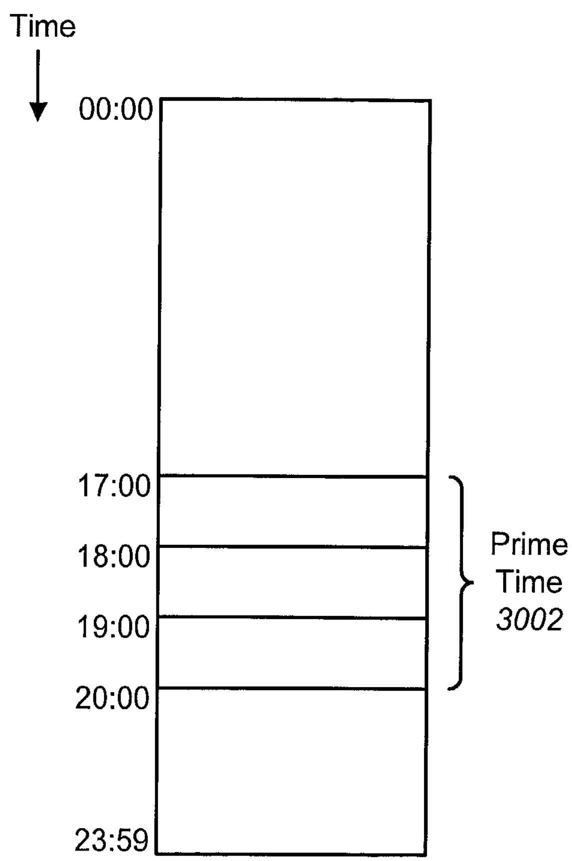


FIG. 30A

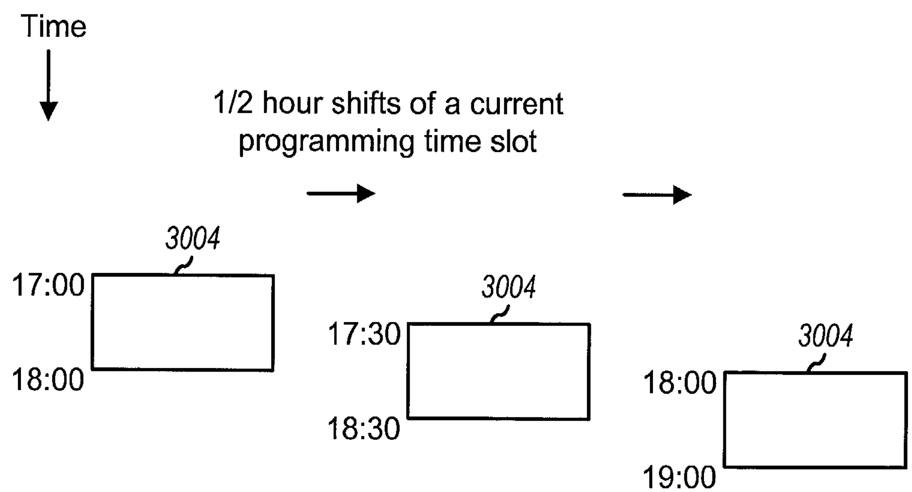


FIG. 30B

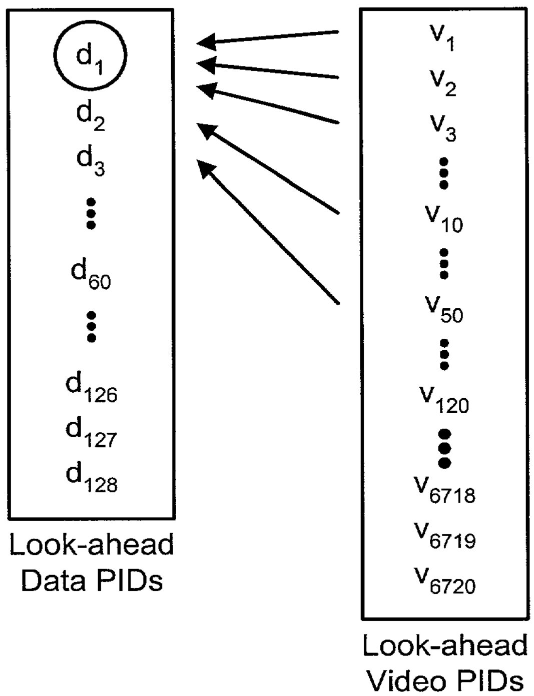

d_1
d_2
d_3
d_{60}
d_{126}
d_{127}
d_{128}
Look-ahead Data PIDs
v_1
v_2
v_3
v_{10}
v_{50}
v_{120}
v_{6718}
v_{6719}
v_{6720}
Look-ahead Video PIDs

FIG. 31

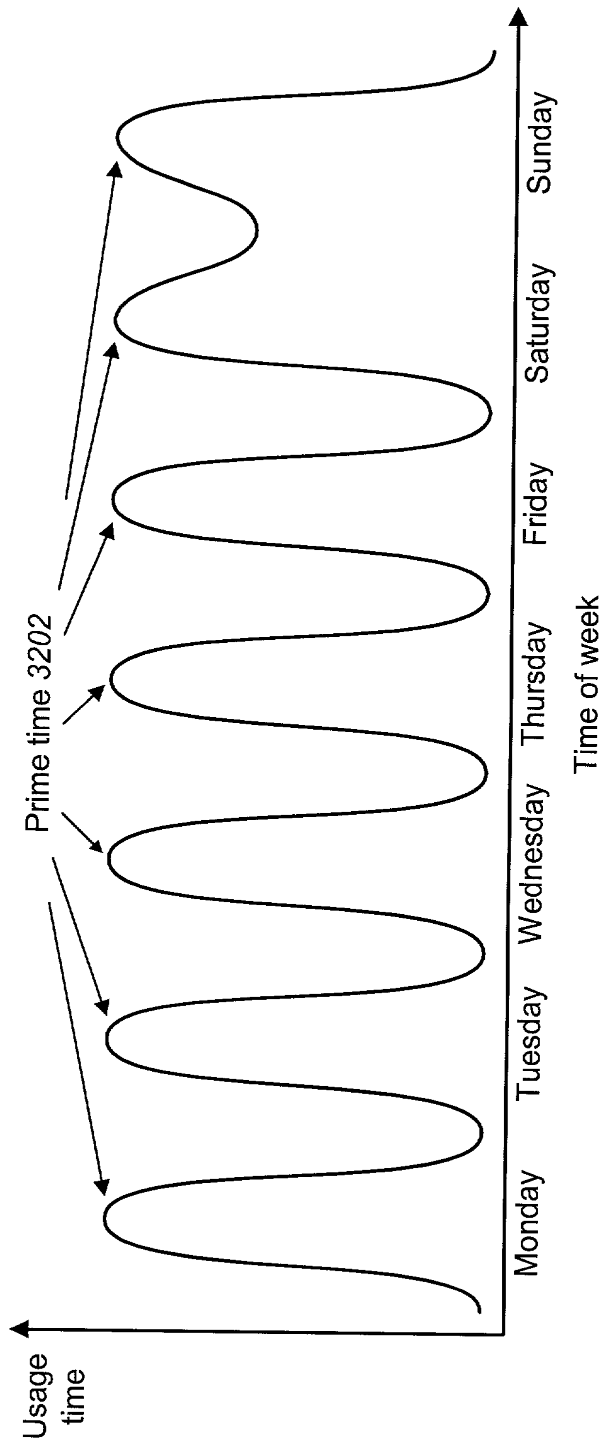

Prime time 3202
Usage time
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Time of week

FIG. 32

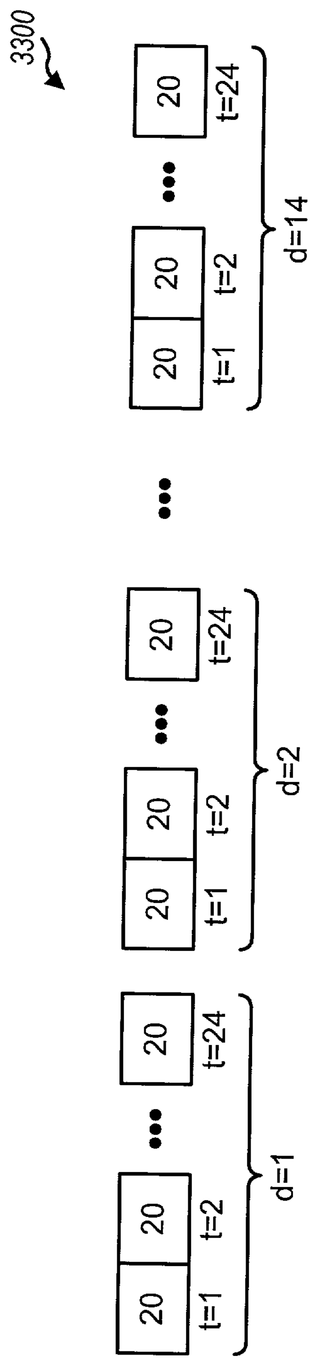

3300
20
20
20
t=1
t=2
t=24
d=1
20
20
20
t=1
t=2
t=24
d=2
20
20
20
t=1
t=2
t=24
d=14

FIG. 33A

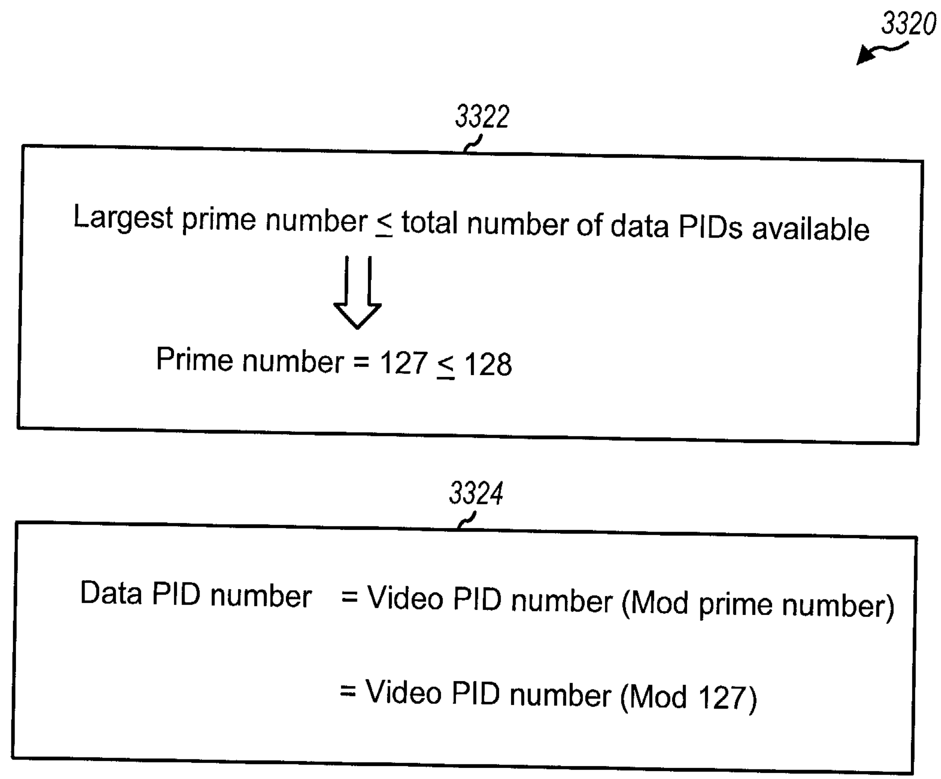
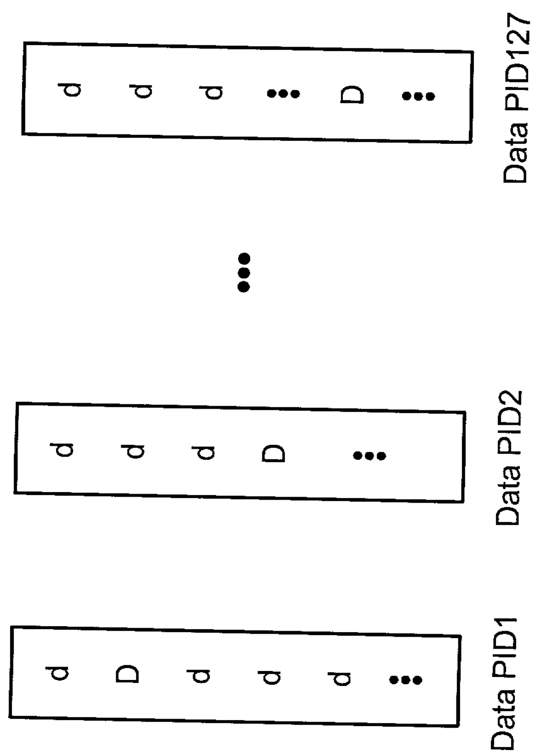

3320
3322
Largest prime number ≤ total number of data PIDs available
Prime number = 127 ≤ 128
3324
Data PID number = Video PID number (Mod prime number)
= Video PID number (Mod 127)

FIG.33B



d = non-prime time data message
D = prime time data message

FIG. 33C

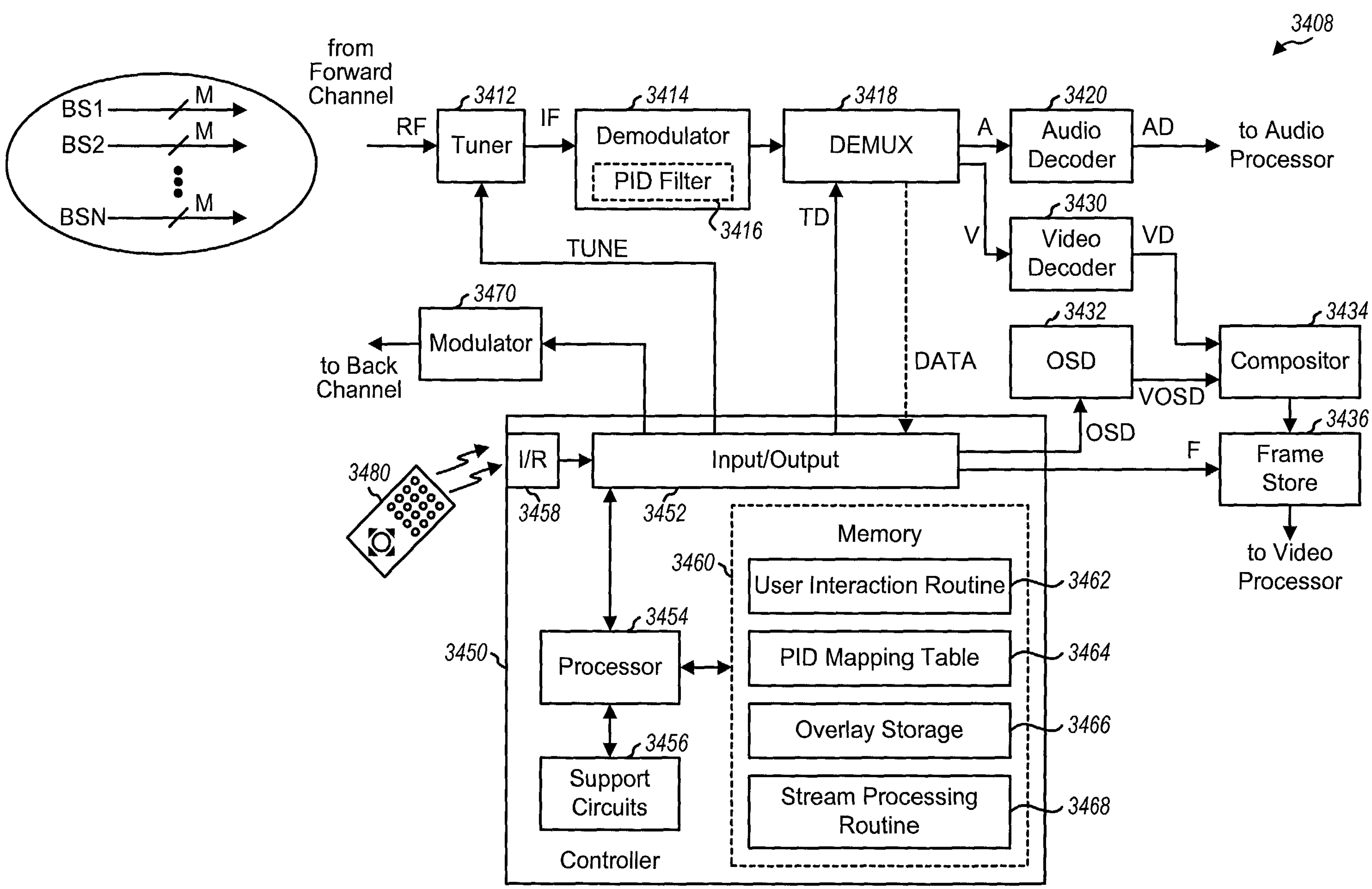

3408
BS1
BS2
BSN
M
M
M
from Forward Channel
RF
3412
Tuner
IF
3414
Demodulator
PID Filter
3416
3418
DEMUX
A
3420
Audio Decoder
AD
to Audio Processor
V
3430
Video Decoder
VD
3432
OSD
VOSD
3434
Compositor
3436
Frame Store
to Video Processor
TD
TUNE
DATA
OSD
F
3470
Modulator
to Back Channel
3480
I/R
3458
Input/Output
3452
3450
3454
Processor
3456
Support Circuits
Controller
Memory
3460
User Interaction Routine
3462
PID Mapping Table
3464
Overlay Storage
3466
Stream Processing Routine
3468

FIG. 34

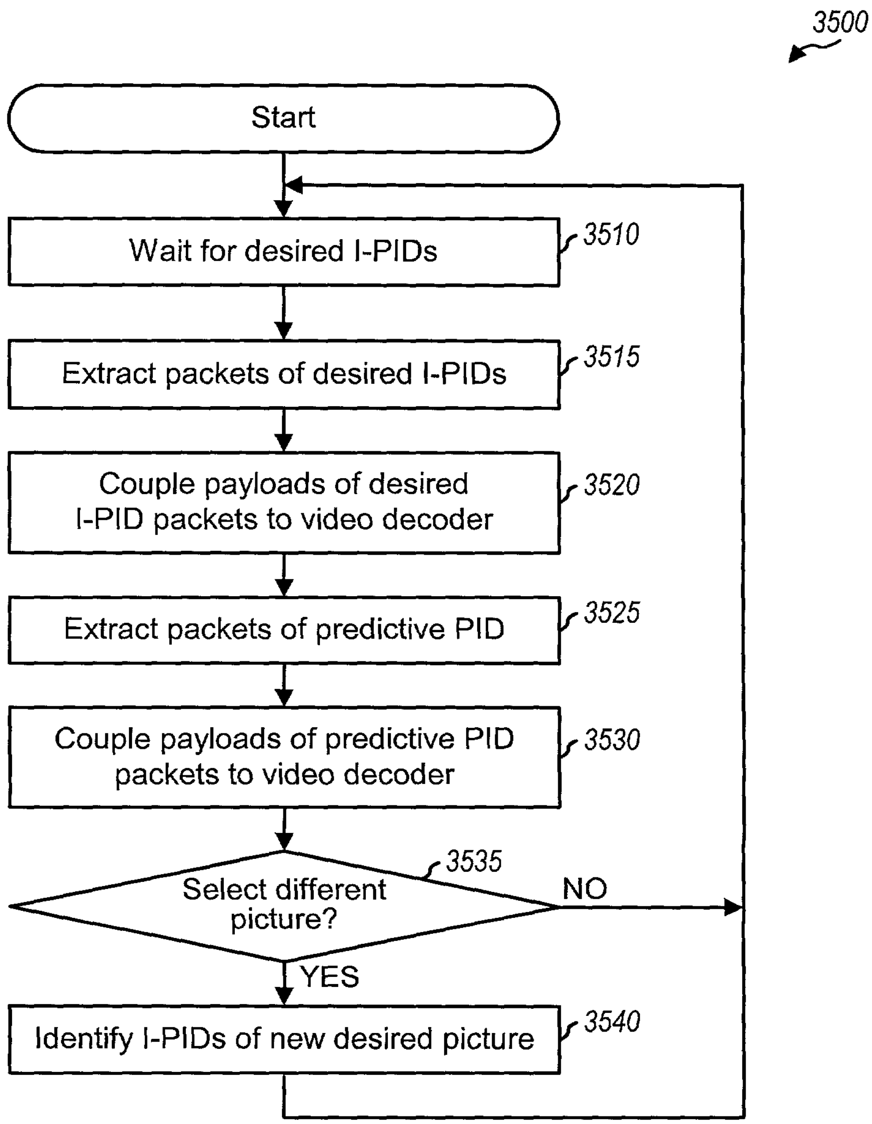

3500
Start
Wait for desired I-PIDs
3510
Extract packets of desired I-PIDs
3515
Couple payloads of desired
I-PID packets to video decoder
3520
Extract packets of predictive PID
3525
Couple payloads of predictive PID
packets to video decoder
3530
Select different
picture?
3535
NO
YES
Identify I-PIDs of new desired picture
3540

FIG. 35

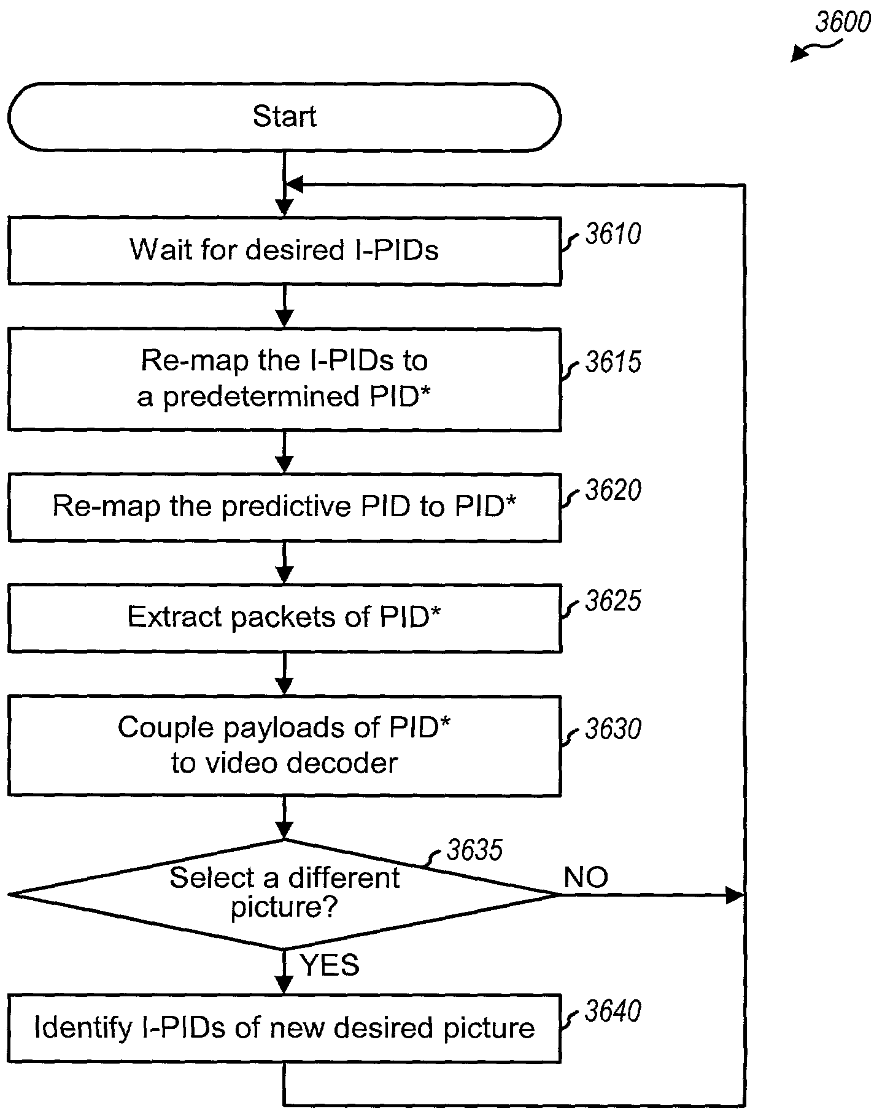

3600
Start
Wait for desired I-PIDs
3610
Re-map the I-PIDs to a predetermined PID*
3615
Re-map the predictive PID to PID*
3620
Extract packets of PID*
3625
Couple payloads of PID* to video decoder
3630
Select a different picture?
3635
NO
YES
Identify I-PIDs of new desired picture
3640

FIG. 36

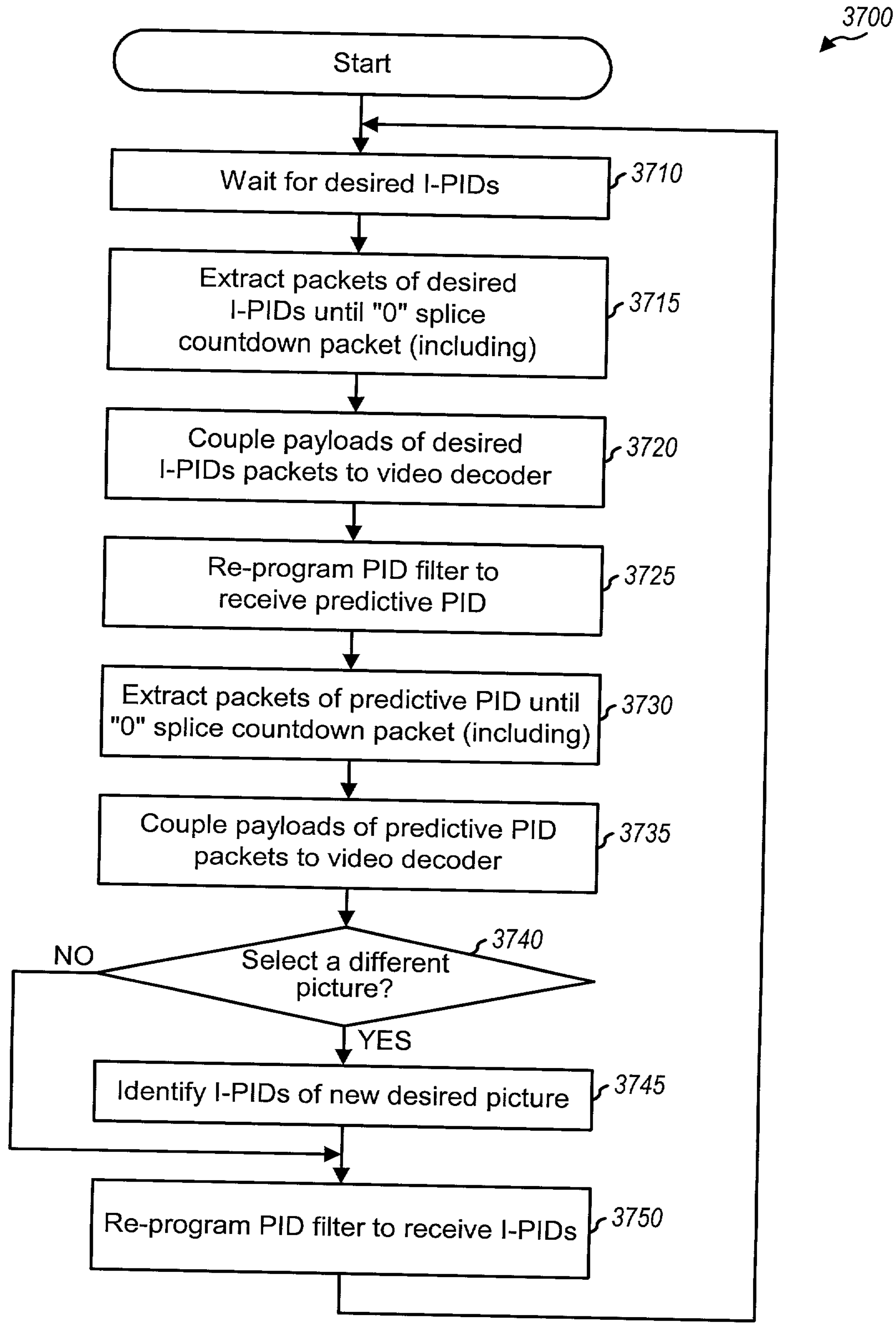

3700
Start
Wait for desired I-PIDs
3710
Extract packets of desired I-PIDs until "0" splice countdown packet (including)
3715
Couple payloads of desired I-PIDs packets to video decoder
3720
Re-program PID filter to receive predictive PID
3725
Extract packets of predictive PID until "0" splice countdown packet (including)
3730
Couple payloads of predictive PID packets to video decoder
3735
Select a different picture?
3740
NO
YES
Identify I-PIDs of new desired picture
3745
Re-program PID filter to receive I-PIDs
3750

FIG. 37

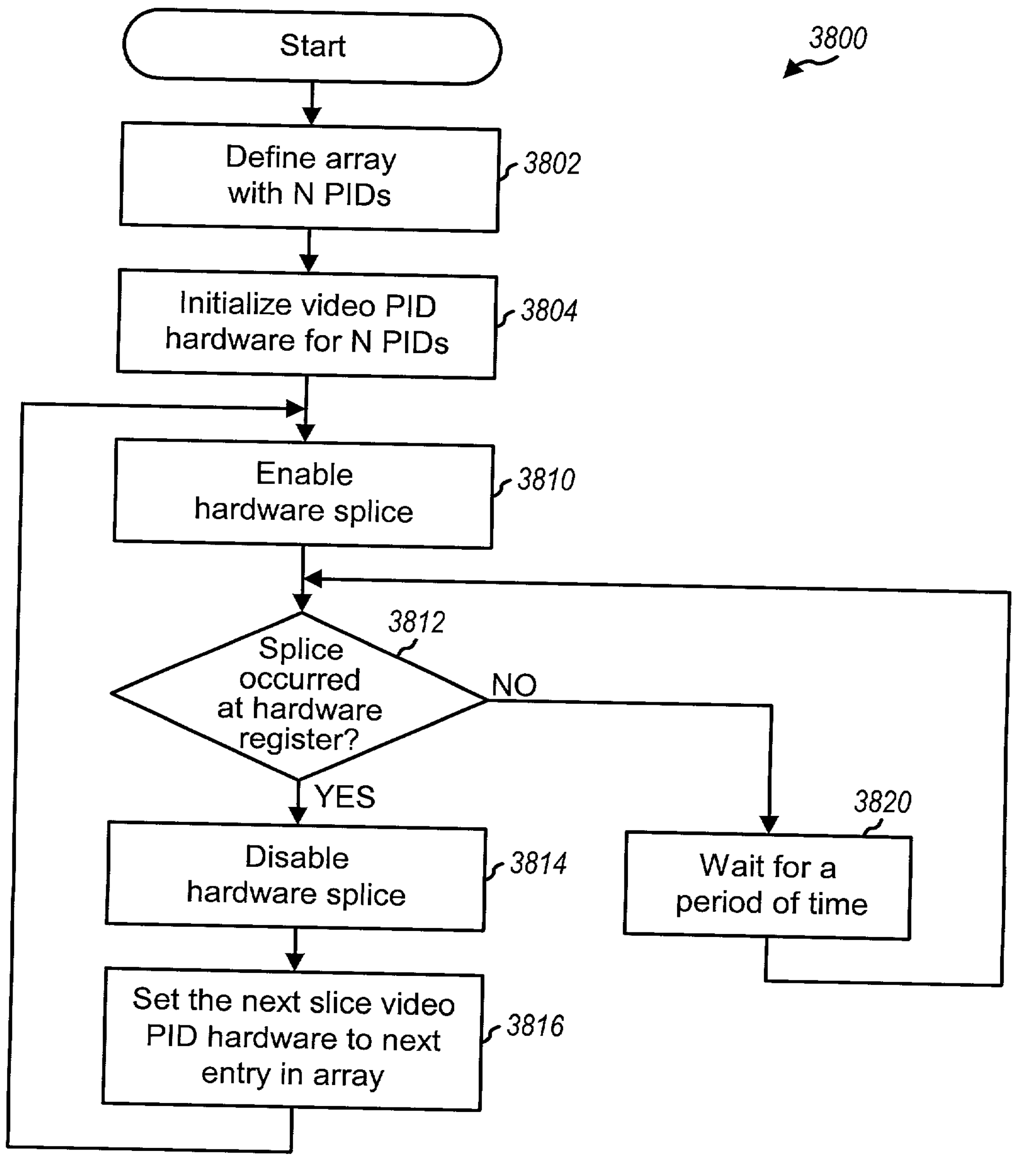

3800
Start
Define array with N PIDs
3802
Initialize video PID hardware for N PIDs
3804
Enable hardware splice
3810
3812
Splice occurred at hardware register?
NO
YES
Disable hardware splice
3814
3820
Wait for a period of time
Set the next slice video PID hardware to next entry in array
3816

FIG. 38

EXHIBIT B

Application Serial No. 09/539,228

Attorney Docket No.: 19880-0016xx
Client Reference No.: 247xx

MESSAGING PROTOCOL FOR DEMAND-CAST SYSTEM AND BANDWIDTH MANAGEMENT

CROSS-REFERENCES TO RELATED APPLICATIONS

This application is based on co-pending U.S. Provisional Patent Application Serial No. 60/178,100, entitled "BANDWIDTH MANAGEMENT TECHNIQUES FOR DELIVERY OF INTERACTIVE PROGRAM GUIDE," filed January 26, 2000. The present application is further a continuation-in-part of co-pending U.S. Patent Application Serial No. 09/524,854, entitled "BANDWIDTH MANAGEMENT TECHNIQUES FOR DELIVERY OF INTERACTIVE PROGRAM GUIDE," filed March 14, 2000. The above-identified related applications are all assigned to the assignee of the present invention and incorporated herein by reference in their entirety for all purposes.

BACKGROUND OF THE INVENTION

The present invention relates to communications systems in general. More specifically, the invention relates to techniques to efficiently deliver interactive program guide (IPG) in a server-centric system.

Over the past few years, the television industry has seen a transformation in a variety of techniques by which its programming is distributed to consumers. Cable television systems are doubling or even tripling system bandwidth with the migration to hybrid fiber coax (HFC) cable plant. Customers unwilling to subscribe to local cable systems have switched in high numbers to direct broadcast satellite (DBS) systems. And, a variety of other approaches have been attempted focusing primarily on high bandwidth digital technologies, intelligent two way set top terminals, or other methods of trying to offer service differentiated from standard cable and over the air broadcast systems.

With this increase in bandwidth, the number of programming choices has also increased. Leveraging off the availability of more intelligent set top terminals, several companies such as Starsight Telecast Inc. and TV Guide, Inc. have developed elaborate systems for providing an interactive listing of a vast array of channel offerings, expanded textual information about individual programs, and the ability to look forward to plan television viewing as much as several weeks in advance.

With this increase in the quantity of programming, it is a challenge to deliver program guide data to viewers in an efficient and effective manner. A large amount of resources (e.g., bandwidth) would normally be needed to continually transmit, for example,

two weeks of programming for 200 channels. Therefore, efficient and effective techniques to deliver interactive program guide to a large number of viewers are highly desirable.

SUMMARY OF THE INVENTION

The present invention provides messaging protocols for use between components of an interactive program guide (IPG) delivery system. The protocol provides a way to more efficiently utilize the finite bandwidth available for distribution of IPG video sequences.

One embodiment of the present invention comprises messaging protocols for communications between a session manager(SM), a transport stream generator (TSG), and a set top terminal (STT). The protocol for communication from the TSG to the STT specifies content for a demand-cast index table that may be transmitted within a private section of a MPEG transport stream. This content includes a list of available demand-cast streams. The protocol for communication from the STT to the SM includes acquisition, release, and request messages. The protocol for communication from the SM to the TSG includes stream release, stream request, and reset messages, and the protocol for communication from the TSG to the SM includes acknowledgements of those messages.

The foregoing, together with other aspects of this invention, will become more apparent when referring to the following specification, claims, and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings.

FIG. 1 is a diagram of an illustrative communications network for distributing video sequences to a number of terminals in accordance with an embodiment of the invention;

FIGS. 2-6 are diagrams of various methods and topologies for demand-casting interactive program guide (IPG) pages in accordance with embodiments of the invention;

FIGS. 2A and 2B are respectively a flow diagram and a topology for a first push method for demand-casting IPG pages in accordance with an embodiment of the invention;

FIGS. 3A and 3B are respectively a flow diagram and a topology for a second push method for demand-casting IPG pages in accordance with an embodiment of the invention;

FIGS. 4A and 4B are respectively a flow diagram and a topology for a first pull method for demand-casting IPG pages in accordance with an embodiment of the invention;

FIGS. 5A and 5B are respectively a flow diagram and a topology for a second pull method for demand-casting IPG pages in accordance with an embodiment of the invention;

FIGS. 6A and 6B are respectively a flow diagram and a topology for a third pull method for demand-casting of IPG pages in accordance with an embodiment of the invention;

FIG. 6C is a flow diagram showing a method of terminating (or continuing) demand-casts in accordance with the third pull method;

FIG. 7 is a diagram of a two-way system for efficient delivery of demand-cast video sequences in accordance with an embodiment of the invention;

FIG. 8 depicts an example of a set of IPG pages for continual broadcast and other IPG pages for demand-cast in accordance with an embodiment of the invention;

FIG. 9 is an example of one picture taken from a video sequence that can be encoded using the invention;

FIGS. 10-13 are block diagrams of first, second, third, and fourth architectures, respectively, for managing delivery of video sequences of an interactive program guide in accordance with embodiments of the invention;

FIGS. 14A-14D are diagrams of an embodiment of the messaging between the terminal, the session manager, and the transport stream generator;

FIG. 15 is a diagram of an example showing the status of active demand-cast streams in an IPG multiplex; and

FIGS. 16A and 16B are diagrams illustrating various scenarios for activation and release of a demand-cast stream.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common within a figure.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

A. ILLUSTRATIVE COMMUNICATIONS NETWORK

FIG. 1 is a diagram of an illustrative communications network 100 for distributing video sequences to a number of terminals in accordance with an embodiment of the invention. Communications network 100 may be a cable distribution network, but other types of distribution networks may also be used and are within the spirit and scope of the invention.

As shown in FIG. 1, communications network 100 includes one or more head-ends (HE) 102, one or more centers for local neighborhood equipment (LNE) 104, a number of distribution nodes 106, and a number of terminals 108. Local neighborhood equipment 104 may be located, for example, at remote hubs of a cable distribution network. Terminals 108 may be user terminals, interactive set-top terminals (STT), or other devices with interactive functionalities.

B. EXAMPLE METHODS AND TOPOLOGIES

As used herein, "demand-cast" refers to the process of managing and delivering content to one or more users in response to user demand for the content. "Broadcast" refers to the process of managing and delivering content to a number of users on a continual basis. "Pointcast" refers to the process of managing and delivering content to a particular user. And "Narrowcast" refers to the process of managing and delivering content to a group of users.

FIGS. 2-6 are diagrams of various methods and topologies for demand-casting interactive program guide (IPG) pages. These methods and topologies are presented for purposes of edification and are not meant to limit the scope of the invention.

1. First Push Method for Demand-cast

FIG. 2A is a flow diagram showing a first push method 200 for demand-casting IPG pages in accordance with an embodiment of the invention. As described below, method 200 includes four steps.

In a first step 202, a first set of IPG pages to be broadcast is predetermined. The first set of IPG pages may comprise video sequences, for example, for a current time

period. For instance, if the current time is 1:07 pm, then the current time period may include programming from 1:00 pm to 2:30 pm, assuming a 90-minute time period.

In a second step 204, a second set of IPG pages to be broadcast is predetermined. The second set of IPG pages may comprise video sequences, for example, for a prime time period. Such prime time period is a time period during which a large number of viewers typically watch TV programming. For example, the prime time period may include programming from 6:00 pm to 9:00 pm.

In a third step 206, the bandwidth to broadcast the first and second sets of IPG pages is allocated by the distribution system for that purpose. For example, as described below in more detail, a bandwidth manager (BWM) within head-end 102 and/or local neighborhood equipment 104 allocates the necessary bandwidth within the in-band network to broadcast the first and second sets of IPG pages to the terminals. If the first and second sets overlap, then only the non-redundant video sequences need to be broadcast, and only enough bandwidth to broadcast the non-redundant video sequences needs to be allocated. Such situation may occur, for example, when the current time period overlaps the prime time period.

In a fourth step 208, the IPG pages of the first and second sets are broadcast to terminals 108 within the broadcast range. The broadcast range may comprise all terminals 108 downstream from head-end 102 or local neighborhood equipment 104. Only non-redundant content needs to be broadcast, and the broadcast is achieved within the allocated in-band bandwidth.

FIG. 2B depicts a first push topology 250 for demand-casting IPG pages in accordance with an embodiment of the invention. Topology 250 corresponds to the first push method 200 of FIG. 2A. As shown in FIG. 2B, the IPG pages are transmitted from head-end 102 or local neighborhood equipment 104 downstream within communications network 100. As shown in FIG. 2B, the broadcast is "pushed" from head-end 102 or line neighborhood equipment 104 to distribution nodes 106 and finally to a number of terminals 108.

2. Second Push Method for Demand-cast

FIG. 3A is a flow diagram showing a second push method 300 for demand-casting IPG pages in accordance with an embodiment of the invention. As described below, method 300 includes three steps.

In a first step 302, one or more IPG pages are selected to be narrowcast to a group of terminals 352. For example, the group of terminals may be a group comprising a high concentration of users with a particular ethnicity or special interest, and the selected IPG page(s) may comprise programming targeted to that ethnic or special interest group. As another example, the group of terminals may comprise terminals in a school campus or business, and the selected IPG page(s) may comprise class instruction or other targeted material. The group of terminals may include terminals in one geographic area or terminals dispersed among different geographic areas but linked, for example, via a network group address.

In a second step 304, the bandwidth to narrowcast the selected IPG page(s) is allocated by the distribution system for that purpose. For example, as described below in more detail, the bandwidth manager (BWM) within head-end 102 and/or local neighborhood equipment 104 allocates the necessary bandwidth within the in-band network to narrowcast the selected IPG page(s) to the group of terminals. If the requested IPG page(s) are already being broadcast as shown in Figs. 2A and 2B, then no additional bandwidth for a narrowcast needs be allocated.

In a third step 306, the selected IPG page(s) are narrowcast to the group of terminals. The narrowcast need only to be received by terminals within the group of terminals 352 and does not need to be received by other terminals. The narrowcast is sent downstream from head-end 102 or local neighborhood equipment 104 to the group of terminals. The narrowcast is achieved within the allocated in-band bandwidth. If the requested IPG page(s) are already being broadcast as shown in Figs. 2A and 2B, then the narrowcast needs not be performed.

FIG. 3B depicts a second push topology 350 for demand-casting IPG pages in accordance with an embodiment of the invention. Topology 350 corresponds to the second push method 300 of FIG. 3A. As shown in FIG. 3B, the IPG page(s) are transmitted from head-end 102 or local neighborhood equipment 104 downstream within communications network 100. As shown in FIG. 3B, the narrowcast is pushed from head-end 102 or line neighborhood equipment 104 to one or more distribution nodes 106 and finally to the terminals within group of terminals 352.

3. First Pull Method for Demand-cast

FIG. 4A is a flow diagram showing a first pull method 400 for demand-casting IPG pages in accordance with an embodiment of the invention. As described below, method 400 includes three steps.

In a first step 402, a request for an IPG page is received from a terminal 108. The request is transmitted upstream from terminal 108 to head-end 102 or line neighborhood equipment 104 via communications network 100. The upstream transmission may be achieved via an out-of-band network or, alternatively, via an in-band network. Such request from the requesting terminal may comprise, for example, a look-ahead request for programming for a time period ahead of the current time period. For a system where one or more IPG pages are already broadcast as shown in Figs. 2A and 2B, the requesting terminal may first check to see whether or not the requested IPG page is already being broadcast before transmitting the request upstream.

In a second step 404, the bandwidth to pointcast the requested IPG page is allocated by the distribution system for that purpose. For example, as described in more detail below, the bandwidth manager within head-end 102 and/or local neighborhood equipment 104 may allocate the necessary bandwidth within the in-band network to pointcast the requested IPG page to the requesting terminal. The allocation is performed if sufficient system resources are available to establish a pointcast session. Moreover, if the requested IPG page is already being broadcast as shown in Figs. 2A and 2B, then no additional bandwidth for a pointcast needs be allocated.

In a third step 406, the requested IPG page is pointcast to the requesting terminal. The pointcast needs only to be received by the requesting terminal and does not need to be received by other terminals. The pointcast is sent downstream from head-end 102 or local neighborhood equipment 104 to the requesting terminal. The pointcast, if necessary, is achieved within the allocated in-band bandwidth. If the requested IPG page is already being broadcast as shown in Figs. 2A and 2B, then the pointcast needs not be performed.

FIG. 4B depicts a first pull topology 450 for demand-casting IPG pages in accordance with an embodiment of the invention. Topology 450 corresponds to first pull method 400 shown in FIG. 4A. As shown in FIG. 4B, the request is transmitted upstream from the requesting terminal to head-end 102 or line neighborhood equipment 104 via communications network 100. Subsequently, the requested IPG page is pointcast

downstream from head-end 102 or line neighborhood equipment 104 to the requesting terminal via communications network 100.

4. Second Pull Method for Demand-cast

FIG. 5A is a flow diagram showing a second pull method 500 for demand-casting IPG pages in accordance with an embodiment of the invention. As described below, method 500 includes three steps.

In a first step 502, a request for an IPG page is received from a requesting terminal 552. The request is transmitted upstream from requesting terminal 552 to head-end 102 or line neighborhood equipment 104 via communications network 100. The upstream transmission may be achieved via an out-of-band network or, alternatively, via an in-band network. Such request may comprise, for example, a look-ahead request for special interest programming available for a future time period ahead of the current time period. For a system where a set or sets of IPG pages are already being broadcast as shown in Figs. 2A and 2B, requesting terminal 552 may first check to determine whether or not the requested IPG page is already being broadcast before transmitting the request upstream.

In a second step 504, the bandwidth to narrowcast the requested IPG page is allocated by the distribution system for that purpose. For example, as described below in relation to Figs. 7 and 8, the bandwidth manager within head-end 102 and/or local neighborhood equipment 104 may allocate the necessary bandwidth within the in-band network to narrowcast the requested IPG page to a group of terminals 554 that includes requesting terminal 552. The allocation is performed if sufficient system resources are available to establish a narrowcast session. If the requested IPG page is already being broadcast as shown in Figs. 2A and 2B, then no additional bandwidth for a pointcast needs be allocated. The group of terminals 554 may include terminals in one geographic area or terminals dispersed among different geographic areas but linked, for example, via a network group address.

In a third step 506, the requested IPG page is narrowcast to group of terminals 554. The narrowcast needs only to be received by the terminals within group of terminals 554 and does not need to be received by other terminals. The narrowcast is sent downstream from head-end 102 or local neighborhood equipment 104 to group of terminals 554. The narrowcast is achieved within the allocated in-band bandwidth. If the requested IPG page is

already being broadcast as shown in Figs. 2A and 2B, then the narrowcast needs not be performed.

FIG. 5B depicts a second pull topology 550 for demand-casting IPG pages in accordance with an embodiment of the invention. Topology 550 corresponds to second pull method 500 shown in FIG. 5A. As shown in FIG. 5B, the request is transmitted upstream from requesting terminal 552 to head-end 102 or line neighborhood equipment 104 via communications network 100. Subsequently, the requested IPG page is narrowcast downstream from head-end 102 or line neighborhood equipment 104 to group of terminals 554, which includes requesting terminal 552, via communications network 100.

5. Third Pull Method for Demand-cast

FIG. 6A is a flow diagram showing a third pull method 600 for demand-casting IPG pages in accordance with an embodiment of the invention. As described below, method 600 includes five steps.

In a first step 602, a request for an IPG page is received from a first terminal 652. The request is transmitted upstream from first terminal 652 to head-end 102 or line neighborhood equipment 104 via communications network 100. The upstream transmission may be achieved via an out-of-band network or, alternatively, via an in-band network. Such request from first terminal 652 may comprise, for example, a look-ahead request for programming for a future time period ahead of the current time period. For a system where one or more IPG pages are already being broadcast as shown in Figs. 2A and 2B, first terminal 652 may first check to see whether or not the requested IPG page is already being broadcast before transmitting the request upstream.

In a second step 604, a stream 656 may be assigned by the distribution system to pointcast the requested IPG page. The assignment is performed if sufficient system resources are available to establish a pointcast session. For example, as described below in more detail, the bandwidth manager within head-end 102 and/or local neighborhood equipment 104 may determine that sufficient resources are available to assign stream 656 to pointcast the requested IPG page to first terminal 652. The stream assignment may be achieved, for example, by assigning a particular value to the program identifier (PID) for stream 656. If the requested IPG page is already being broadcast as shown in Figs. 2A and 2B, then stream 656 needs not be assigned.

In a third step 606, the requested IPG page is pointcast to first terminal 652 via assigned stream 656. This may be achieved by transmitting packets that are identified by the particular PID value and contain a video sequence of the requested IPG page. The pointcast needs only to be received by first terminal 652 and does not need to be received by other terminals. The pointcast is sent downstream from head-end 102 or local neighborhood equipment 104 to first terminal 652. If the requested IPG page is already being broadcast as shown in Figs. 2A and 2B, then the pointcast needs not be performed.

In a fourth step 608, a request for an IPG page is received from a second terminal 654. In this example, the IPG page requested by second terminal 654 is the same as the IPG page requested by first terminal 652. Similar the first request, the second request is transmitted upstream from second terminal 654 to head-end 102 or line neighborhood equipment 104 via communications network 100 via an out-of-band network or an in-band network. Second terminal 654 may be in the same or different geographic area as first terminal 652.

In a fifth step 610, the identifier (e.g., PID value) for the assigned stream 656 is transmitted from head-end 102 or line neighborhood equipment 104 to second terminal 654. This enables the next step 612 to occur without use of additional PIDs or additional network bandwidth.

And in a sixth step 612, second terminal 654 receives the requested IPG page via the same assigned stream 656, which was used to deliver the IPG page to first terminal 652. Second terminal 654 may be set to decode and present packets that are identified by the particular PID value for stream 656. Such packets contain the video sequence of the requested IPG page. In this manner, "sharing" of stream 656 occurs, changing the previous "single" pointcast to a "double" pointcast.

Similarly, other terminals may "share" the pointcast if they request the same IPG page and can receive the requested IPG page via the same stream 656. In this manner, any number of terminals may share the pointcast. This sharing results in more efficient use of the available bandwidth.

FIG. 6B depicts a third pull topology 650 for demand-casting IPG pages in accordance with an embodiment of the invention. Topology 650 corresponds to pointcast "sharing" method 600 shown in FIG. 6A. As shown in FIG. 6B, a request is transmitted upstream from first terminal 652 to head-end 102 or line neighborhood equipment 104 via communications network 100. In response, the requested IPG page is pointcast by stream

656 from head-end 102 or line neighborhood equipment 104 to first terminal 652. Next, a second request for the same IPG page is transmitted upstream from second terminal 654 to head-end 102 or line neighborhood equipment 104 via communications network 100. In response, the identifier for stream 656 is transmitted from head-end 102 or line neighborhood equipment 104 to second terminal 654. Subsequently, second terminal 654 uses the identifier to receive the IPG page from the same stream 656.

FIG. 6C is a flow diagram showing a method 660 of terminating (or continuing) demand-casts in accordance with third pull method 600. As described below, method 660 includes five steps.

In a first step 662, a terminal finishes viewing a stream used to send an IPG page. In the example described above in Figs. 6A and 6B, the terminal may be either first terminal 652 or second terminal 654. In general, the terminal may be any of the terminals that are sharing the same stream, or the last terminal to view a stream that was previously shared.

In a second step 664, head-end 102 or line neighborhood equipment 104 is notified that the terminal has finished viewing the stream. Such notification can be achieved by the terminal by sending a communication upstream to head-end 102 or line neighborhood equipment 104 via an out-of-band or in-band network.

In a third step 666, a determination is made whether or not that stream is being viewed by one or more terminals. As described in more detail below, this determination is done within head-end 102 or line neighborhood equipment 104 and may be done by a bandwidth manager in conjunction with a session manager.

In a fourth step 668, if one or more terminals are still viewing that stream, then head-end 102 or line neighborhood equipment 104 continues to transmit the stream. Such transmission is typically performed by an in-band delivery system.

Finally, in a fifth step 670, if no other terminals are viewing that stream, then the stream is "torn down" so that it is no longer transmitted and no longer takes up network bandwidth. The torn down stream is then available for reassignment and the bandwidth can be reused to transmit a different pointcast, narrowcast, or broadcast.

C. DEMAND-CAST SYSTEM

1. Guide Page Usage Frequency Distribution

The usage of guide pages can be characterized by their frequency distribution. Certain pages in a guide page matrix, such as those in the current time slot and adjacent time slots ("near look-ahead") are likely to be accessed more frequently by viewers. Other guide pages, such as the "far look-ahead" pages, are likely to be accessed less frequently. These characteristics of guide page usage can be supported by a demand-cast model described herein. Access to all possible guide pages in the guide page matrix can be achieved by sending in a transport stream a combination of continually broadcast guide pages for pages that are more frequently accessed, and temporarily broadcast or demand-cast guide pages for pages less frequently accessed. In an embodiment, current and near look-ahead pages are sent in a broadcast fashion and far look-ahead pages are sent in a demand-cast fashion.

2. Demand-cast Overview

A demand-cast IPG system is a two-way system employing communication between the terminal in the communications network and the head-end via a back-channel. Demand-cast pages are inserted in the transport stream for temporary broadcast in response to access demand generated by viewers in the network. When a particular viewer request a particular guide page, one of two things can occur. If the requested page is already in the IPG broadcast, the terminal simply acquires the corresponding stream. Otherwise, if the page is not in the broadcast, the terminal requests the head-end to insert a stream in the IPG multiplex for the requested page. The head-end can then replace the least frequently accessed and not currently accessed stream in the IPG multiplex with a stream for the newly requested page.

When a terminal no longer accesses a guide page, it informs the head-end that it has released it. When accessing a demand-cast page, an IPG application at the terminal can time-out following a certain particular period of inactivity (e.g., 2 minutes) by the viewer. If a time-out occurs, the terminal can inform the head-end that it has released the page. Informing the head-end when demand-cast pages are released ensures that non-accessed demand-cast pages are available for substitution. If a terminal requests a new demand-cast page to be inserted into the IPG multiplex and there is no slot available in the IPG multiplex, the head-end refuse to insert a stream for the newly requested guide page, which then results in a blockage. Most statistical multiplexed systems are susceptible to blockage if loaded with an excessive number of users and during chaotic episodes. An advantage of the demand-cast

model is that if a particular page is likely to be extensively accessed, such as a page listing a major sports event, the page only needs to be inserted once into the transport stream. The page is then readily accessible by any number of terminals without consuming additional bandwidth.

3. Latency in Broadcast vs. Demand-cast

Access to guide pages within a short delay (i.e., with low latency) is an important feature for interactive program guide. Continually broadcast pages offer a low latency access, whereas demand-cast pages may incur additional processing delays if not yet included in the transport stream. In an embodiment, frequently accessed pages, such as those in the current time slot and near look-ahead time slots, and perhaps prime-time slots are broadcast continually so that they can be accessed with the lowest possible latency. Less frequently accessed far look-ahead pages can be sent via demand-cast.

4. System Description

FIG. 7 is a diagram of a two-way system 700 that can efficiently deliver demand-cast video sequences in accordance with an embodiment of the invention. System 700 includes a session manager (SM) 702 and a transport stream generator (TSG) 704.

Session manager 702 and transport stream generator 704 may be co-located within a distribution center. The distribution center may comprise, for example, head-end 102 in communications network 100. Alternatively, session manager 702 and transport stream generator 704 may be at different locations. For example, session manager 702 may be located at head-end 102, and transport stream generator 704 may be located at local neighborhood equipment 104 in communications system 100.

Session manager 702 and transport stream generator 704 are both coupled to a number of terminals 708 via a distribution network. The distribution network may comprise, for example, a cable distribution network as illustrated in FIG. 1. In that example, terminals 708 would comprise terminals 108 or an equivalent functionality integrated into a computer system or advanced television. Alternatively, for example, the distribution network may comprise a satellite communications system or another type of communications system (telephonic, wireless, etc.).

One terminal 708 and its links to session manager 702 and transport stream generator 704 are illustrated in FIG. 10. In the specific embodiment shown in FIG. 10,

terminal 708 receives in-band communication from transport stream generator 704 and sends out-of-band (OOB) communications to session manager 702. In an alternative embodiment, the communication to session manager 702 may comprise upstream in-band communications.

Session manager 702 may comprise, in one embodiment, a computer system residing at head-end 102. The computer system may comprise, for example, a computer server running a particular operating system (e.g., a version of the UNIX or Windows operating system). The computer system may receive out-of-band communication from terminals 708 via a connection to the network controller. For example, the connection may comprise an Ethernet connection, and the network controller may comprise a controller from General Instruments Corp (now part of Motorola Inc.) or another supplier. The computer system also communicates with and controls transport stream generator 704 via a network connection such as an Ethernet connection.

Session manager 702 manages delivery of IPG pages to terminals 708 on a number of cable nodes, with each node being served by a separate IPG multiplexed transport stream generated at a corresponding transport stream generator 704. Session manager 702 also monitors demand-cast stream usage by terminals 708. Session manager 702 tracks IPG demand-cast streams that are acquired by at least one terminal 708. For example, session manager 702 can maintain a table that dynamically lists which terminals 708 are using each stream. This tracking is performed for each IPG multiplexed transport stream managed by session manager 702.

Session manager 702 also accepts messages from terminals 708 indicating that they have acquired, released, or requested demand-cast streams. A new terminal 708 that has acquired a demand-cast stream is registered (i.e., added) to the stream, and a terminal 708 that has released a demand-cast stream is removed from the registry for the stream. Session manager 702 informs the corresponding transport stream generator 704 if there is no longer any terminals 708 registered to a particular demand-cast stream, and also informs transport stream generator 704 when a terminal 708 requests a demand-cast stream. In one embodiment, session manager 702 may time-out acquisition of a stream by any terminal 708 if the terminal has not released the stream within a particular period of time (e.g., a few minutes). The time-out may be implemented by removing terminals 708 from the registry for the stream after the particular period of time.

Transport stream generator 704 may comprise, in one embodiment, a computer system residing at head-end 102. The computer system may comprise, for

example, a computer server running a particular operating system (e.g., a version of Windows or UNIX operating system). Alternatively, transport stream generator 704 may be located apart from session manager 702, for example, at local neighborhood equipment 104. Each transport stream generator 704 is coupled to an associated session manager 702, for example, via an Ethernet network. Transport stream generator 704 may generate one or more IPG multiplexed transport stream, with each transport stream being broadcast to a respective node in the distribution system.

In one embodiment, the IPG multiplexed transport stream comprises an MPEG transport stream. In this case, transport stream generator 704 may communicate with terminals 708 via tables in the private section of the MPEG transport stream. Such table may include a list of available demand-cast streams, along with the address of the source transport stream generator 704 and information to identify the particular IPG multiplexed transport stream to which the table belongs.

Transport stream generator 704 manages each IPG multiplexed transport stream that it generates. Transport stream generator 704 receives information from session manager 702 indicating whether each demand-cast stream being served is currently being acquired by any terminal, or not at all. That is, transport stream generator 704 is informed by session manager 702 when a demand-cast stream is no longer being acquired any terminals 708.

In one embodiment, transport stream generator 704 maintains a status for each demand-cast stream being served. The status for each stream is adjusted upon receipt by transport stream generator 704 of certain messages from session manager 702. In an embodiment, the basic states for the stream status comprise an "acquired" state that denotes that the demand-cast stream is being acquired by one or more terminals 708, and a "released" state that denotes that the demand-cast stream is not being acquired by any terminal 708. Transport stream generator 704 continues to serve "acquired" demand-cast streams by multiplexing them into the appropriate transport streams and replaces "released" demand-cast streams with new demand-cast streams upon receipt of request messages from session manager 702. In an embodiment, transport stream generator 704 also keeps track of the order in which the streams are released, so that the oldest released stream may be used as the most likely candidate for replacement.

If all demand-cast streams in a particular IPG multiplexed transport stream are "acquired," then a new stream may not be inserted into the transport stream, and transport

stream generator 704 may refuse any new requests. In such case, a message indicating such refusal may be sent to session manager 702 and/or the requesting terminal 708.

In an embodiment, terminal 708 comprises a set-top terminal (STT) for use by a service subscriber. The STT may comprise an embedded system that includes a tuner, a demultiplexer, and a decoder, as described in further detail below. The STT drives the subscriber's display unit or TV set, and it may be coupled to transport stream generator 704 via an RF feed from a cable distribution network. The IPG pages may be received from a particular IPG multiplexed transport stream on a particular modulated carrier signal. In an embodiment, the IPG multiplexed transport stream may comprise an ensemble of elementary MPEG video streams, with each elementary stream representing a portion of the guide.

In an embodiment, terminal 708 includes IPG client software application that resides at the terminal. The IPG client application is responsible for presenting the IPG to the viewer, and demultiplexes and decodes IPG pages requested by the user. If a requested page is already being received via the IPG multiplexed transport stream, then the IPG client application acquires the stream immediately and sends a message to session manager 702 indicating that it has acquired the stream. And if the requested page is not in the IPG multiplexed transport stream, then the IPG client application sends a request message to session manager 702. Subsequently, the IPG client application acquires the stream once it is transmitted by transport stream generator 704 and received by terminal 708. In addition, if a stream is no longer being acquired, the IPG client application sends a release message to session manager 702. In an embodiment, if there is no remote control or other activity by the user for a particular period of time (e.g., a few minutes), then the IPG client application times-out the acquisition. The time-out may be accomplished, for example, by sending a release message to session manager 702 and acquiring a broadcast stream instead.

D. MAJOR MODULES OF DEMAND-CAST SYSTEM

The demand-cast system includes three major subsystems: the set top terminal (STT), the session manager (SM), and the transport stream generator (TSG). For a better understanding of the invention, a specific implementation of each subsystem is now described. Other implementations are also possible and within the scope of the invention.

1. Set-Top Terminal (STT)

The STT is the end-user or cable service subscriber tuner/demultiplexer/decoder and embedded system. In an embodiment, the STT used in initial pilot deployments of the demand-cast system is the General Instruments DCT-2000. The STT is coupled to the cable operator RF feed and drives the subscribers display unit or TV set. The IPG content is provided in an IPG transport stream (i.e., IPG multiplex) located on a specific QAM carrier. The IPG multiplex contains an ensemble of elementary MPEG video streams, with each elementary video stream representing portions of the guide and some of these streams representing guide grid pages. The STT receives messages from the head-end via tables in the private section of the IPG transport stream (in-band messaging.) The STT sends messages to the head-end via an out-of-band back-channel or return path.

The STT includes an IPG application that is responsible for presenting (e.g., the DIVA Interactive Program Guide) to the viewer. The IPG application demultiplexes and decodes IPG pages requested by the user. If a particular page is in the IPG transport stream, the STT can quickly acquire the stream inform the session manager that it has requested the page. And if the page is not in the IPG multiplex, the STT also sends a message to the session manager that it has requested it. The STT then acquires the stream once the stream is included in the IPG multiplex. When the STT no longer acquires a particular guide stream, it informs the session manager that it has released the stream.

In an embodiment, if the STT is on a particular demand-cast stream for more than a particular period of time (e.g., 2 minutes) without any remote control activity, the STT times-out. The STT then acquires a broadcast stream instead and informs the session manager that it has released the demand-cast stream.

2. Session Manager (SM)

In an embodiment, the session manager is implemented with a computer system (e.g., a SPARC Station running the Solaris operating system from SunMicrosystems, Inc.) residing at the cable head-end. The session manager is coupled via Ethernet to the server side of a network controller (NC) from General Instruments Corp. and is the receiver for out-of-band return path messages originating from the STTs. The session manager can handle STTs on multiple cable nodes, each node being served by a separate IPG multiplex. The session manager communicates with and controls the transport stream generators via Ethernet. The transport stream generators generate the IPG transport streams.

The session manager manages one or more cable networks and monitors demand-cast stream usage. The session manager also tracks IPG demand-cast streams that are acquired by at least one STT and maintains a dynamic list of STTs that are using each demand-cast stream. This tracking is achieved for each IPG multiplex managed by the session manager. The session manager accepts messages from the STTs indicating requests for, or release of, demand-cast streams. An STT that has acquired a demand-cast stream is registered to the stream, and an STT that has released a demand-cast stream is removed from the stream's registry. The session manager informs the transport stream generator if there are no longer any STTs using a particular demand-cast stream, and also informs the transport stream generator when an STT requests a demand-cast stream.

In an embodiment, the session manager times-out an STT from a demand-cast stream if the STT has not released the stream within a particular time period (e.g., a few minutes). The session manager can achieve this by removing the STT from the demand-cast stream's registry.

3. Transport Stream Generator (TSG)

In an embodiment, the transport stream generator is implemented with a computer system (e.g., running a WindowNT operating system from Microsoft Corp.) residing at the cable head-end. The transport stream generator is coupled via Ethernet to the session manager controlling it. The transport stream generator produces one or more IPG transport streams, with each transport stream being broadcast to a respective node. In an embodiment, the transport stream generator communicates with the STTs via tables in the private section of the IPG transport streams. The table contains a list of the available demand-cast streams along with the IP address of the source transport stream generator (e.g., its IP address) and the channel number of the IPG multiplex (i.e., which multiplex it is in the transport stream generator).

The transport stream generator manages the transport stream for each IPG multiplex it generates. The transport stream generator receives information from the session manager for each demand-cast stream indicating whether the stream is currently acquired by any STT or released by all STTs. The transport stream generator is informed by the session manager when there is no longer any STT on a particular demand-cast stream and when an STT requests a demand-cast stream.

The transport stream generator maintains status for all demand-cast streams in each IPG multiplex. The transport stream generator adjusts the status for each demand-cast stream each time it receives a message from the session manager for the stream. The basic status for each stream includes "acquired" for a stream that is in use by one or more STTs and "released" for a stream that is not in use by any STT. The transport stream generator continues to send "acquired" streams in its IPG multiplexes and replaces "released" streams with new demand-cast streams as they are requested. The transport stream generator also keeps track of the age of the released streams and the best candidate for replacement is the oldest released stream. If all demand-cast streams in a multiplex are "acquired" then it may not be possible to insert a new stream when requested and the transport stream generator can refuse to process the request.

E. EXAMPLE OF INTERACTIVE PROGRAM GUIDE

An embodiment of an interactive program guide in accordance with the invention is described below. The embodiment is described for purposes of illustration and is not meant to limit the scope of the invention.

FIG. 8 depicts an example of a set of IPG pages for continual broadcast and other IPG pages for variable demand-cast in accordance with an embodiment of the invention. In the specific example shown in FIG. 8, 40 IPG pages are continually broadcast and up to 30 IPG pages may be variably demand-cast. There are 10 guide pages per time slot, and the continual broadcast includes 10 guide pages for the current time slot and 30 guide pages for the next three 1-hour time slots. The variably demand-cast pages may be any pages within the guide page matrix that are not currently being broadcast.

In such a system, when a request for a guide page is made by a particular terminal, either one of two scenarios can occur. First, if the page is already in the IPG broadcast, then the terminal simply acquires the stream for the page from the IPG broadcast. Alternatively, if the page is not in the broadcast, then the terminal transmits a request for the page to the head-end. The head-end may then fulfill the request by replacing the currently transmitted stream that is least frequently accessed and not currently accessed with another stream containing the requested page.

Subsequently, the terminal eventually ends its access of the guide page. This may occur because the user has instructed the terminal to view a different page. Alternatively, this may occur because of a time-out due to inactivity over a particular period

of time (e.g., 2 minutes). In any case, if the terminal is no longer accessing the guide page, then the terminal transmits a message to the head-end indicating that it has released the corresponding stream. Informing the head-end when demand-cast pages become released ensures that non-accessed demand-cast pages become available for substitution, as described above.

An advantage of the invention is that, if a particular page is extensively accessed (such as a page listing a major sports event), then the system needs to insert the particular page only once into the transport stream. Once inserted, the page is readily accessible by any number of terminals without requiring additional bandwidth. Other systems may be more susceptible to blockage, which occurs, for example, when a newly requested page cannot be inserted into the transport stream because no bandwidth is available within the transport stream.

An IPG delivery system in accordance with an embodiment of the invention is a two-way system that is capable of supporting two-way communication between the terminals on the cable network and the equipment in the cable head-end. For example, communication may be transmitted from the terminals to the head-end via a back-channel, and content may be transmitted from the head-end to the terminals by insertion into a transport stream.

FIG. 9 depicts an example of an IPG page 900 in accordance with an embodiment of the invention. In the specific embodiment shown in FIG. 9, IPG page 900 includes a time slot region 905, a guide region 910, a video region 920, an icon region 940, a program description region 950, a logo region 960, and a date/time display 970. Other designs for the IPG page with different layouts, configurations, and combinations of regions and objects can be contemplated and are within the scope of the invention.

Time slot region 905 includes a first time slot object 905a and a second time slot object 905b that indicate the time slots for which program guide is being provided on the IPG page. Guide region 910 is used to display program listings for a group of channels. In the embodiment shown in FIG. 9, the program listings show the available programming in two half-hour time slots. Guide region 910 thus includes a number of channel objects 912a through 912j used to display channel information for a guide listing of channels. Guide region 910 further includes a pair of channel indicator icons 914a and 914b that identifies the current cursor location.

Program description region 950 is used to present descriptive information relating to a particular program selected from the program listings, or may be used to show other information. Video region 920 may be used to display images, videos, text, or a combination thereof, which may be used for advertisements, previews, or other purposes. Video region 920 may be implemented as described above in a server-centric manner. Logo region 960 may include a logo of a service operator or other entity and may be optionally displayed. Date/time display 970 may be configurable by the user and may also be optionally displayed.

Icon region 940 is used to display various icons, which may be created and/or enabled by the user. Each icon in icon region 940 can represent a filter or a link to another IPG page or a particular interface. Each filter selects a particular type of programming to be included in the program listings shown in guide region 902. For example, a Pay Per View (PPV) icon 941 may be a filter that selects only PPV programming to be included in the program listings. A Favorites icon 942 may be a filter that selects only channels designated by the user to be among his or her favorites. A Movies icon 943 may be a filter that selects only movies or movie channels. A Kids icon 944 may be a filter that selects only channels for children or programming appropriate for or produced for viewing by children. A Sports icon 945 may be a filter that selects only sports channels or sports-related programming. A Music icon 946 is a link to a music interface. An Options icon 947 may also be a link to a menu of IPG options that the user may select amongst. The options may include (1) configuration and selection/deselection information of IPG related services, (2) custom information such as deactivating some of the filters or accessing the custom condensed listing menus, and others. A Weather icon 948 may be a link to an interface to weather information.

In a system, illustratively, comprising 80 channels of information, the channels are displayed in 10-channel groups having associated with them two half-hour time slots. In this organization, 8 video PIDs are provided to carry the present-time channel/time/title information, one or more audio PID is provided to carry the audio barker and/or one or more data PIDs (or other data transport method) are provided to carry the program description data, overlay data, and the like. To fully broadcast interactive program information for up to 24 hours in advance, 192 (e.g., 8*24) video PIDs are provided, along with one or more audio PIDs and, optionally, one or more data PIDs.

The time depth of a program guide is defined by the amount of time in programming is provided in the broadcast video PIDs for the particular channel groups. The

channel depth of the program guide is defined by the number of channels available through the guide (as compared to the total number of channels in the system). In a system providing only half of the available channels via the broadcast video PIDs, the channel depth 50%. In a system providing 12 hours of "look-ahead" time slot, the time depth is 12 hours. In a system providing 16 hours of time slot "look-ahead" and 4 hours of "look-back" time slot, the time depth is +16/-4 hours.

The video streams representing the IPG are sent in a one or more transport streams, within the form of a single or multi-programs as described above. A user desiring to view the next 1-hour time interval (e.g., 10:00 – 11:00) may activate a "scroll right" object (or move the joystick to the right when a program within guide region 910 occupies the final displayed time interval). Such activation results in a controller within the terminal noting that a new time interval is desired. The video stream desired to the new time interval is then decoded and displayed. If the corresponding video stream is within the same transport stream (i.e., a new PID), then the stream is simply decoded and presented. If the desired video stream is within a different transport stream, then that transport stream is extracted from the broadcast stream and the desired video stream is decoded and presented. And if the desired transport stream is within a different broadcast stream, then that broadcast stream is tuned, the desired transport stream is extracted, and the desired video stream is decoded and presented.

A user interaction requesting in a prior time interval or a different set of channels results in the retrieval and presentation of the desired video stream. If the desired video stream is not part of the broadcast video streams, then a pointcast session, for example, may be initiated as described above for Figs. 4A and 4B. For this pointcast session, the terminal sends a message to the head-end via a back channel requesting a particular stream. The head-end processes the request, retrieves the desired stream from the information server, and incorporates the stream within a transport stream as a video PID. Preferably, the desired stream is inserted into the transport stream currently being tuned/selected by the terminal. The head-end further informs the terminal which PID should be received and from which transport stream it should be demultiplexed. The terminal then retrieves the desired video PID. If the video PID is within a different transport stream, the terminal first demultiplexes that transport stream (possibly by tuning a different QAM stream within the forward channel).

Typically, upon completion of the viewing of the desired stream, the terminal indicates to the head-end that it no longer needs the stream. In response, the head-end tears down the pointcast session. The terminal then returns to the broadcast stream from which the pointcast session was launched. However, as described above in Figs. 6A, 6B, and 6C, the method for "sharing" pointcasts may delay or avoid the need to tear down the pointcast session if another terminal is still utilizing the pointcast. In addition, the above described pointcast sharing technique more efficiently utilizes the network bandwidth allocated for pointcasts.

Push demand-casts and pull demand-casts are associated with different delays (i.e., latencies). Access to IPG pages with low latency is a desirable feature in interactive program guide. A system that only pushes IPG pages may be able to offer access with the lowest possible latency, whereas a system that only pulls pages may incur significant processing delays in accessing each page.

In accordance with an embodiment of the invention, more frequently accessed IPG pages such as those in the current time slot and near look-ahead time slots, and perhaps prime-time slots are push demand-cast continually so that access can be achieved with low latency. Less frequently accessed (e.g., far look-ahead) pages are pull demand-cast.

F. EXAMPLE IMPLEMENTATIONAL ARCHITECTURES

Four architectures are described below for delivery of interactive program guide. These architectures are illustrative of the architectures that may be used to implement various aspects of the invention. However, other architectures may also be used and are within the scope of the invention.

FIG. 10 is a diagram of a first architecture 1000 for managing delivery of video sequences of an interactive program guide in accordance with an embodiment of the invention. First architecture 1000 includes a key manager 1003, a subscription/billing manager 1004, an IPG generator 1006, and a head-end 1002. First architecture 1000 is capable of providing encryption for the IPG content.

Head-end 1002 couples to a number of terminals 1008 via an in-band network and/or an out-of-band (OOB) network. Head-end 1002 includes various elements that couple together and interact with each other to provide the desired functionality. In the embodiment shown in FIG. 10, head-end 1002 includes an advertising/HTML content source 1007, an IPG content source 1009, a compositor 1010, an encoder 1012, a processor 1014, a multiplexer

1016, an encryptor 1018, an in-band delivery system 1020, a controller 1022, a session manager 1024, an access manager 1026, a bandwidth manager 1028, and an out-of-band (OOB) equipment 1030.

It is noted that session manager 702 in FIG. 7 encompasses the functionality of a number of elements in FIG. 10, including session manager 1024 and bandwidth manager 1028. Also, it is noted that transport stream generator 704 in FIG. 7 also encompasses the functionality of a number of elements in FIG. 10, including processor 1014 and multiplexer 1016.

FIG. 11 is a diagram of a second architecture 1100 for managing delivery of video sequences of an interactive program guide in accordance with an embodiment of the invention. Second architecture 1100 includes the elements in first architecture 1000. In addition, second architecture 1100 includes local neighborhood equipment 104 and a video-on-demand (VOD) server 1005. Second architecture 1100 is also capable of providing encryption for the IPG content.

As shown in FIG. 11, line neighborhood equipment 1004 couples to head-end 1002 via an in-band network and an out-of-band messaging system. Line neighborhood equipment 1004 also couples to a number of terminals 1008 via a local in-band network. Line neighborhood equipment 1004 includes various elements that couple together and interact with each other to provide the desired functionality. Line neighborhood equipment 1004 typically includes a subset of the type of components in head-end 1002. In the embodiment shown in FIG. 11, line neighborhood equipment 1004 includes a processor 1114, a multiplexer 1116, an encryptor 1118, a local delivery system 1120, a controller 1122, a session manager (SM) 1124, an access manager (AM) 1126, and a bandwidth manager (BWM) 1128.

FIG. 12 is a diagram of a third architecture 1200 for managing delivery of video sequences of an interactive program guide in accordance with an embodiment of the invention. Third architecture 1200 includes a local IPG center 1204, a head-end 1202, a service center 1206, and a number of terminals 1208. In addition, the system may be integrated with a video-on-demand (VOD) system 1210 and a corresponding VOD application 1238 at terminal 1208. Third architecture 1200 does not support encryption of the IPG content.

Local IPG center 1204 generates guide page user interface (UI) screens and periodically sends the UI screens to an IPG server 1212 at head-end 1202. A multiple service

operator (MSO)/third party IPG add-on content 1214 may be provided to IPG server 1212 from MSO equipment within head-end 1202. For example, the add-on content may include real-time advertisement video or HTML pages for electronic commerce.

IPG server 1212 composes (C), encodes (E), processes (P), multiplexes (M), and modulates (QAM) the IPG content (guide plus add-on content) and sends it to a combiner 1216. Combiner 1216 combines the IPG content with broadcast TV, premium content (e.g., HBO), pay-per-view (PPV), and other content from a multiple service operator (MSO) content delivery system 1218. The combined content is then broadcast to terminals 1208 via an in-band distribution network 1220.

Upon viewer tuning of terminal 1208 to an IPG channel, an IPG application 1222 at the terminal processes the IPG stream and provides the IPG via an application programming interface (API) 1224 to a "native" application 1226 running on the terminal. Native application 1226 decodes and presents the IPG to the viewer.

In one embodiment, the TV program guide for a current time period (1-hour) is broadcast to viewers. In addition, two weeks of look-ahead TV programming may be delivered to viewers on demand via demand-cast. Upon a viewer action of moving a cursor to a look-ahead time interval, the terminal sends a request via a back-channel to a session manager (SM) 1228 (e.g., via an out-of-band channel to a reverse path demodulator (RPD), then to a network controller (NC), then to session manager 1228). Session manager 1228 then causes IPG server 1212 to multiplex the requested IPG page into the IPG stream.

Session manager 1228 also interacts with a subscriber/billing interface 1230 in VOD system 1210 to coordinate access to VOD services from a link in the IPG user interface. The user interface also provides for access to premium content and pay-per-view purchasing by interaction through a two-way interface to an MSO customer management system (CMS) 1232 and digital access controller (DAC) 1234 in service center 1206. DAC 1234 generates digital encryption-related keys.

Third architecture 1200 also includes a bandwidth manager (BWM) 1236. As described above, bandwidth manager 1236 provides techniques to more efficiently utilize the limited bandwidth available for distribution of the IPG.

It can be noted that session manager 702 of FIG. 7 encompasses the functionality of a number of elements in FIG. 12, including session manager 1228 and bandwidth manager 1236. It can also be noted that transport stream generator 704 in FIG. 7

encompasses the functionality of a number of elements in FIG. 12, including the processor (P) and the multiplexer (M).

FIG. 13 is a diagram of a fourth architecture 1300 for managing delivery of video sequences of an interactive program guide in accordance with an embodiment of the invention. Fourth architecture 1300 in FIG. 13 is similar to third architecture 1200 in FIG. 12 and also does not support encryption of the IPG content.

Fourth architecture 1300 differs from third architecture 1200 primarily in that some of the elements are distributed from head-end 1202 to a number of hubs 1304 in the distribution system. In particular, combiner 1216, processor (P), multiplexer (M), and modulator (QAM) are moved from head-end 1202 to each hub 1304. Thus, the functionality of transport stream generator 704 is transferred to hubs 1304.

G. MESSAGING PROTOCOL

A specific messaging protocol for communicating between the major components of the system is now described in relation to FIG. 14A through 14D. Other messaging protocols can also be used and are within the scope of the invention.

In an embodiment, the "source" transport stream generator communicates with a terminal via, for example, a one-way in-band channel. The communication may be, for example, in the form of a "demand-cast index table" within a private section of the IPG MPEG transport stream.

FIG. 14A depicts an embodiment of the content of a demand-cast index table. The content may include: (a) a table version number (which increments when the table content changes); (b) a list of available demand-cast streams; (c) an internet protocol (IP) address for the source transport stream generator; (d) a MUX channel number within the source transport stream generator, and (e) a time of day and day of week.

In an embodiment, the terminal communicates with the session manager via, for example, a one-way out-of-band return path. The return path may be implemented, for example, using a user datagram protocol/internet protocol (UDP/IP) service to connect the terminal to a network controller (NC) at the session manager.

FIG. 14B depicts an embodiment of the contents of a message sent from the terminal to the session manager. The message content as shown includes: (a) a demand-cast stream identification; (b) the terminal's identification; (c) the IP address of the source transport stream generator; (d) the MUX channel number within the source transport stream

generator; and (e) the message information itself. The message information may indicate: (1) an acquisition of the demand-cast stream by the terminal; (2) a release of the demand-cast stream by the terminal; or (3) a request for the demand-cast stream by the terminal.

In an embodiment, the session manager communicates with the source transport stream generator via, for example, a two-way communications channel. The two-way communications channel may comprise, for example, a TCP/IP connection over an Ethernet network.

FIG. 14C depicts an embodiment of the contents of a message sent from the session manager to the transport stream generator. The message content as shown includes: (a) the demand-cast stream identification; (b) the MUX channel number within the source transport stream generator; and (c) a particular message/command selected from a set of possible messages/commands. The set of messages/commands include: (1) demand-cast stream released (no longer acquired by any terminal); (2) demand-cast stream requested; and (3) a reset command.

Messages from the session manager to the transport stream generator may be acknowledged by the transport stream generator.

FIG. 14D depicts an embodiment of the contents of an acknowledgement message sent by the transport stream generator back to the session manager. An acknowledgement message as shown includes: (a) the demand-cast stream ID; (b) the MUX channel number; (c) the transport stream generator's IP address; and (d) the acknowledgement itself. The acknowledgement may acknowledge (1) release of the demand-cast stream; (2) request for the demand-cast stream; or (3) reset of the transport stream generator.

H. STREAM STATUS AND CONVERSIONS OF STATUS

The following relate to stream status and conversions of status in accordance with a specific embodiment of the invention. Other stream statuses and conversions of status can also be implemented and are within the scope of the invention.

1. Stream Status Within IPG Multiplex

The transport stream generator models bandwidth usage for each IPG multiplexed transport stream that it is managing. Each demand-cast stream within each transport stream may be either active or inactive. Active streams are currently being

multiplexed into the transport stream, and inactive streams are not currently being multiplexed into the transport stream.

FIG. 15 depicts an example showing statuses of a number of active demand-cast streams in an IPG multiplex within a transport stream generator. For each demand-cast stream, the transport stream generator assigns a status with respect to the stream's intended multiplex. Demand-cast stream statuses, in context of the transport stream generator, are:

1) "active" streams are in the IPG multiplex

    1.1) "acquired" demand-cast streams are in the multiplex but are used by at least one terminal. They are referenced in the demand-cast index table in the private section of the IPG transport stream. They are not candidates for removal.

    1.2) "released" demand-cast streams are in the multiplex and are not used by any terminal. They are referenced in the demand-cast index table. They can become "passive".

        1.2) "passive" demand-cast streams are also technically "released". They are in the multiplex but are not used by any terminal. They are not referenced in the demand-cast index table. They are typically a small fraction of the "released" demand-cast streams. A few oldest 'released' demand-cast streams are forced to the "inactive" status by a maintenance thread. They are candidates for removal.

2) "inactive" demand-cast streams are not in the IPG multiplex. They are not referenced in the demand-cast index table. They may be inserted in the multiplex

Note that the transport stream generator may remove all the "passive" demand-cast streams from their respective IPG multiplexes and replace them with null packets. It may be advantageous to leave "passive" demand-cast streams in the multiplex in case a terminal requests it, in which case the latency will be minimized.

2. Conversions of Status

The transport stream generator receives messages from the session manager. Messages received from the session manager are:

1) "request demand-cast stream"

2) "release demand-cast stream" The "release demand-cast stream" message includes the maximum number of terminals that were registered to the demand-cast stream.

3) "reset"

A. Transport Stream Generator Request Demand-cast Stream

If the transport stream generator receives a "request demand-cast stream" message from the serial number, then the following methods for activating the stream are possible. FIG. 15 illustrates the various methods for activating a demand-cast stream.

1) If the demand-cast stream is currently "inactive", then

a) In a first case, if there are one or more "passive" demand-cast streams in the corresponding multiplex, then the transport stream generator removes a "passive" demand-cast stream from the multiplex, and replaces it with the new requested demand-cast stream. The transport stream generator adds reference to the new 'active' demand-cast stream in the demand-cast index table. The transport stream generator assigns the status 'active' to the newly inserted demand-cast stream. The transport stream generator acknowledges the session manager for the "request demand-cast stream" message by sending a "success" message back to the session manager.

b) In a second case, if there are no "passive" demand-cast streams in the corresponding multiplex, but a 'released' demand-cast stream is included therein, then the transport stream generator forces the oldest 'released' demand-cast stream to the "inactive" status and then follows the steps described above for the first case.

c) In a third case, if the transport stream generator finds no "passive" or "released" demand-cast stream in the corresponding multiplex, it can not fulfill the request. The transport stream generator acknowledges the session manager for the "request demand-cast stream" message by sending a "fail" message back to the session manager.

2) If the demand-cast stream is currently 'released' or 'passive', then

a) The transport stream generator changes the status of the 'released' or 'passive' demand-cast stream to 'acquired.' The transport stream generator also acknowledges the session manager for the "request demand-cast stream" message by sending a "success" message back to the session manager.

B. Transport Stream Generator Release Demand-cast Stream

If the transport stream generator receives a "release demand-cast stream" message from the session manager, then it acknowledges the session manager by sending a "success" message. If the demand-cast stream is currently 'acquired', then the transport stream generator changes the status of the stream to 'released.'

C. Released Stream to Passive Stream Conversion

Removal of a 'released' demand-cast stream can be done. However, such removal, unless necessary, may be disadvantageous. Initially, the reference to the 'released' demand-cast stream is removed from the "demand-cast index table", then a short time later (e.g., few seconds) later the stream can be physically removed from the multiplex. This delay between the removal from the table and the removal from the multiplex prevents a race condition whereby a terminal is acquiring a demand-cast stream while the transport stream generator is in the process of removing it. Therefore, only 'passive' streams are removed in accordance with an embodiment of the invention.

The ratio of 'passive' to 'released' demand-cast stream may be specified in the transport stream generator configuration file. It may be maintained as a percentage (e.g., 10% of 'released' streams are converted to 'passive') or it can be maintained as an absolute number (e.g., so as to ensure that there are usually two or three 'inactive' demand-cast streams).

FIG. 20 illustrates an overall process by which a released stream may be converted to a passive stream. Methods for determining which released streams are converted to passive streams include an aging method and a statistical method. In the aging method, the few oldest 'released' demand-cast streams are continually converted to 'passive' status by a maintenance thread. In the statistical method, the "release demand-cast stream" messages include statistical data regarding the demand-cast stream. This data may provide the maximum number of terminals that were on a released stream before it was released. The transport stream generator converts demand-cast streams that have had the least amount of users to 'passive' status.

I. OTHER TECHNICAL ASPECTS

The following are further technical aspects in accordance with a specific embodiment of the invention. Other variations are also possible and within the scope of the invention.

1. Initial Conditions

Set Top Terminal : When the STT launches the IPG application, it tunes to the QAM carrying the IPG transport stream. When the STT requests its first demand-cast stream, it opens the IPG channel with the session manager. When the QAM is tuned, the STT acquires the demand-cast index table and sends an "Init" command to the session manager.

Session Manager : Initially, the session manager has no knowledge of the IPG multiplex fed to its client STTs. Upon receiving a first "request demand-cast stream" message from a STT, the session manager verifies that it is aware of the MUX ID. MUX ID includes the transport stream processor IP address and MUX channel within the transport stream generator. If the session manager is aware, then nothing happens. And if the session manager is not aware, the transport stream generator opens a communication socket with the source transport stream generator. The session manager maintains a log where it registers all MUXes that it controls. For each MUX in the log, the transport stream generator's IP address and MUX channel number is recorded.

Transport Stream Generator : Initially, the transport stream generator is configured through its own configuration file. Configuration includes the number of demand-cast streams that can be supported by each IPG multiplex. The absolute number of 'passive' streams or the ratio of 'passive' streams to 'released' streams is specified in the configuration file

2. Reset

Set Top Terminal : When the STT does not "see" the PID of the demand-cast stream it is acquiring in the demand-cast index table, it acquires a default IPG broadcast PID. If the STT does not see the demand-cast index table, the STT exits the IPG application.

Session Manager : If the session manager is down, upon reset, it looks up transport stream generator log file and sends a reset command to the transport stream generator.

Transport Stream Generator : When the transport stream generator receives a "Reset" command from the session manager, it removes reference to all demand-cast streams in the demand-cast index table in the multiplex referenced by the MUX ID in the reset command. Then a short time (e.g., a few second) later, the transport stream generator removes all the demand-cast streams within the multiplex.

3. Scalability

Set Top Terminal : STT messages regarding demand-cast streams include demand-cast stream ID, transport stream generator's IP address, and the MUX channel number on the transport stream generator.

Session Manager : The session manager can manage more than one transport stream generator. Each IPG multiplex is referred to by the IPG address of the host transport stream generator and the MUX channel number on the transport stream generator.

Transport Stream Generator : Each transport stream generator can manage more than one IPG multiplex.

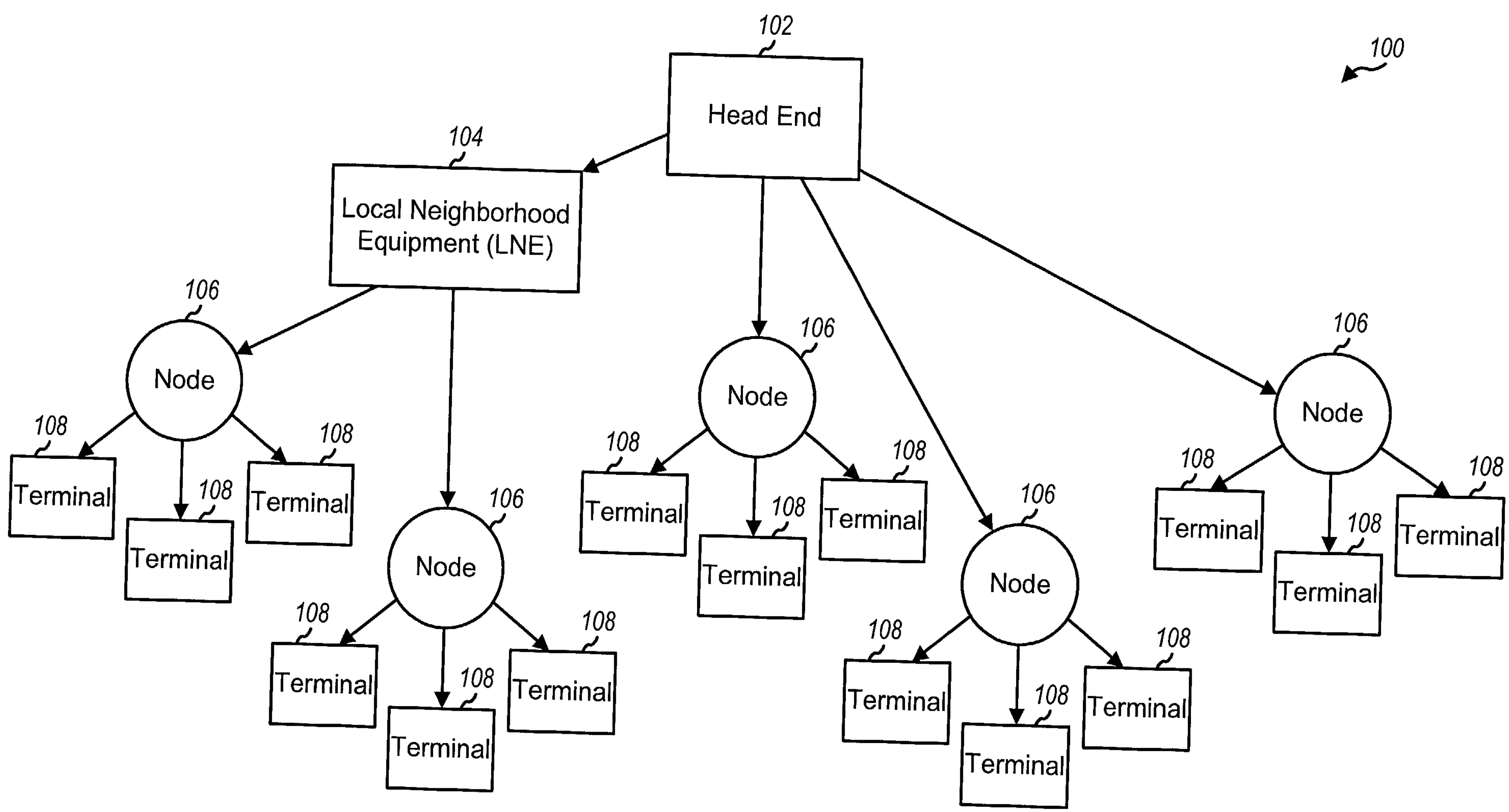
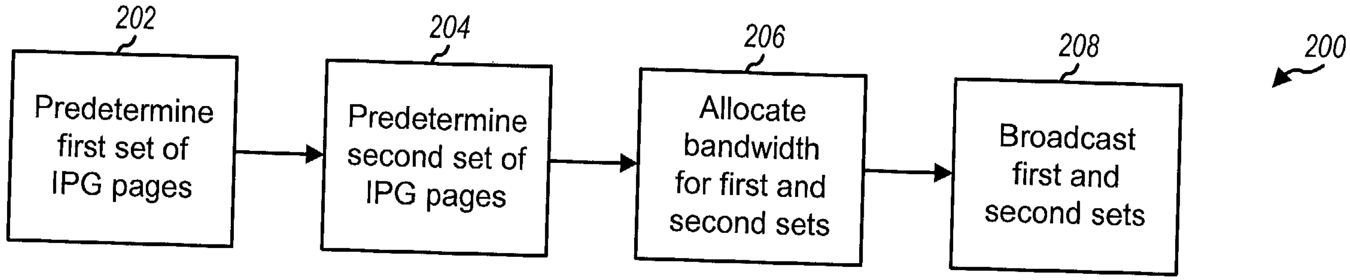
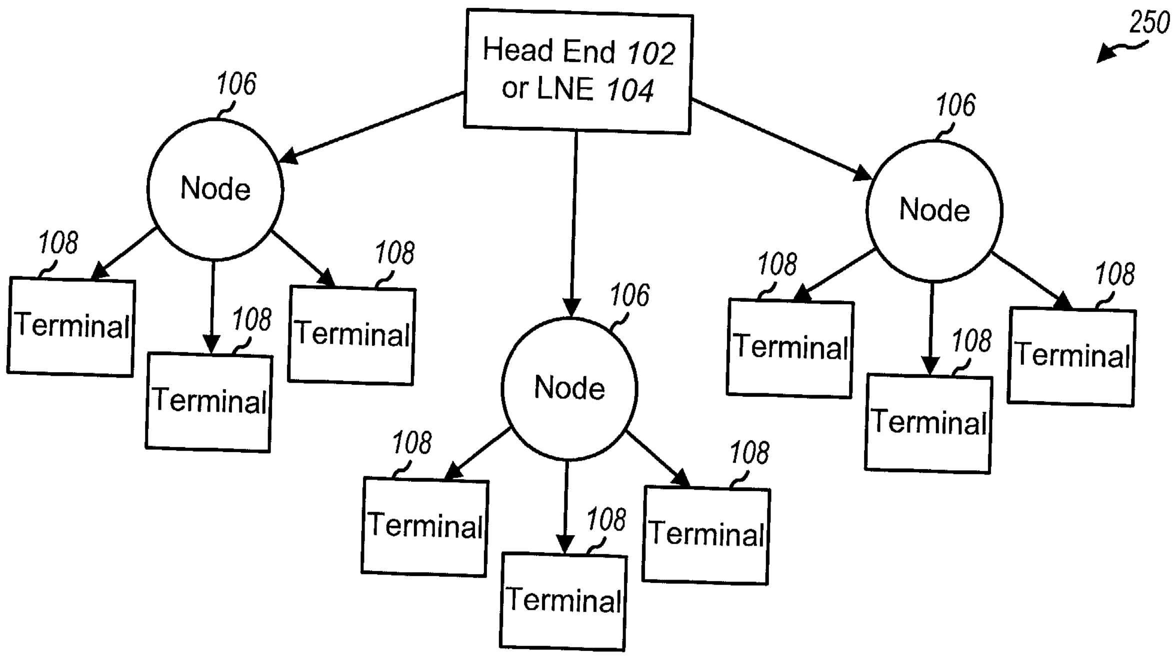

100
102
Head End
104
Local Neighborhood Equipment (LNE)
106
Node
108
Terminal

FIG. 1



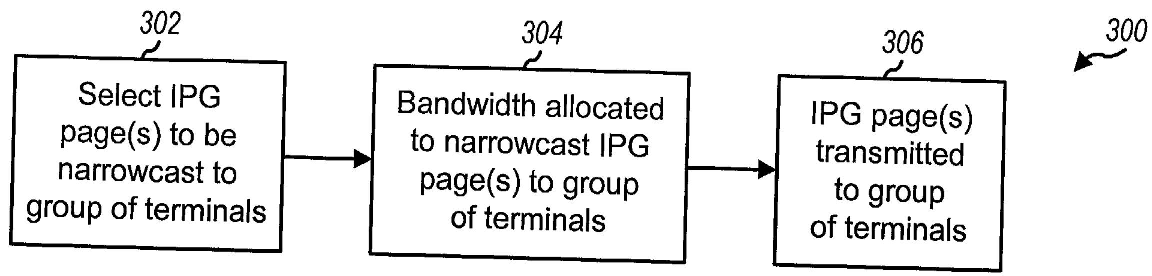
First push demand-cast (broadcast) method

FIG. 2A



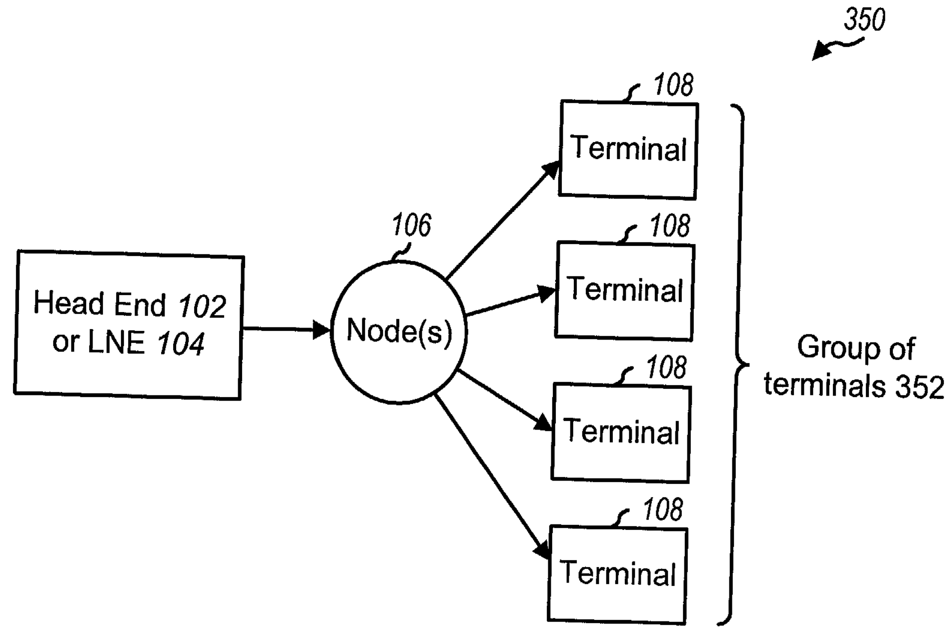
First push demand-cast (broadcast) topology

FIG. 2B



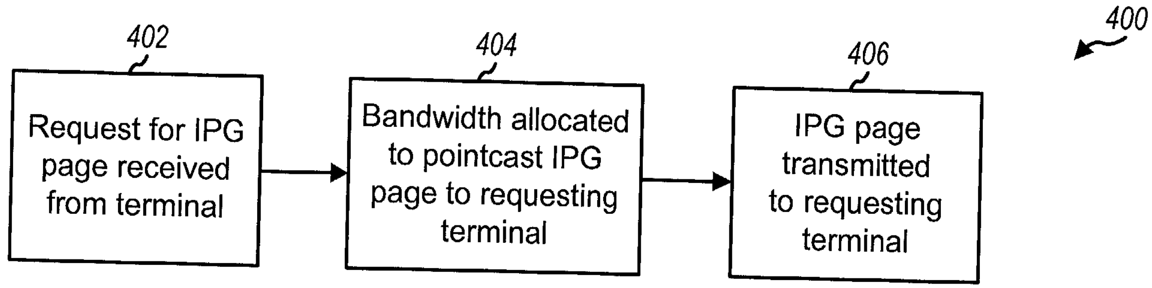
Second push demand-cast (narrowcast) method

FIG. 3A



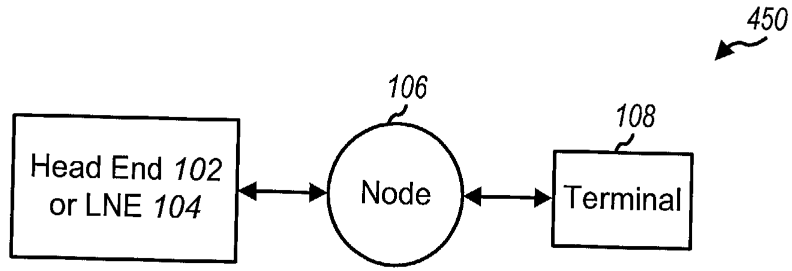
Second push demand-cast (narrowcast) topology

FIG. 3B



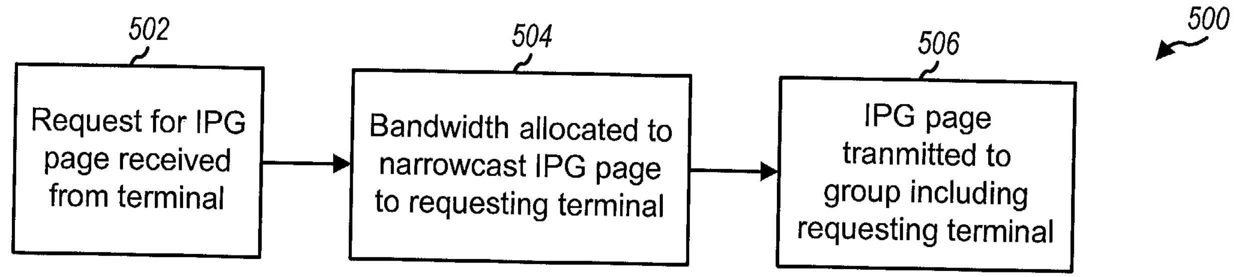
First pull demand-cast (pointcast) method

FIG. 4A



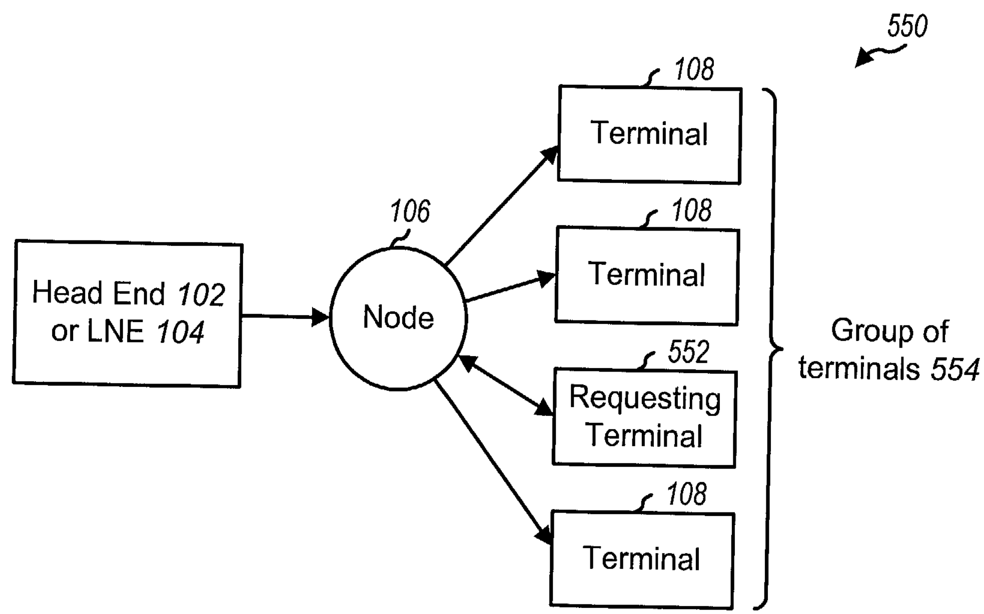
First pull demand-cast (pointcast) topology

FIG. 4B



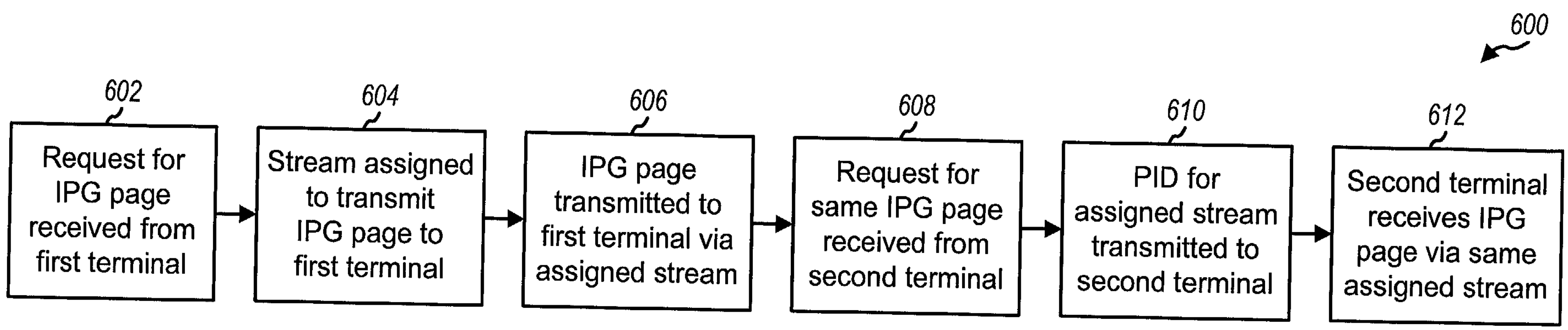
Second pull demand-cast (narrowcast) method

FIG. 5A



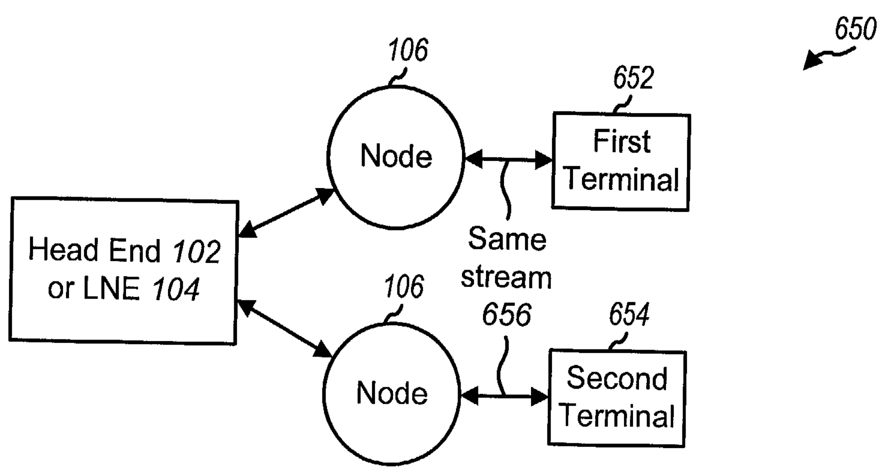
Second pull demand-cast (narrowcast) topology

FIG. 5B



Third pull demand-cast (pointcast sharing) method

FIG.6A



Third pull demand-cast (pointcast sharing) topology

FIG. 6B

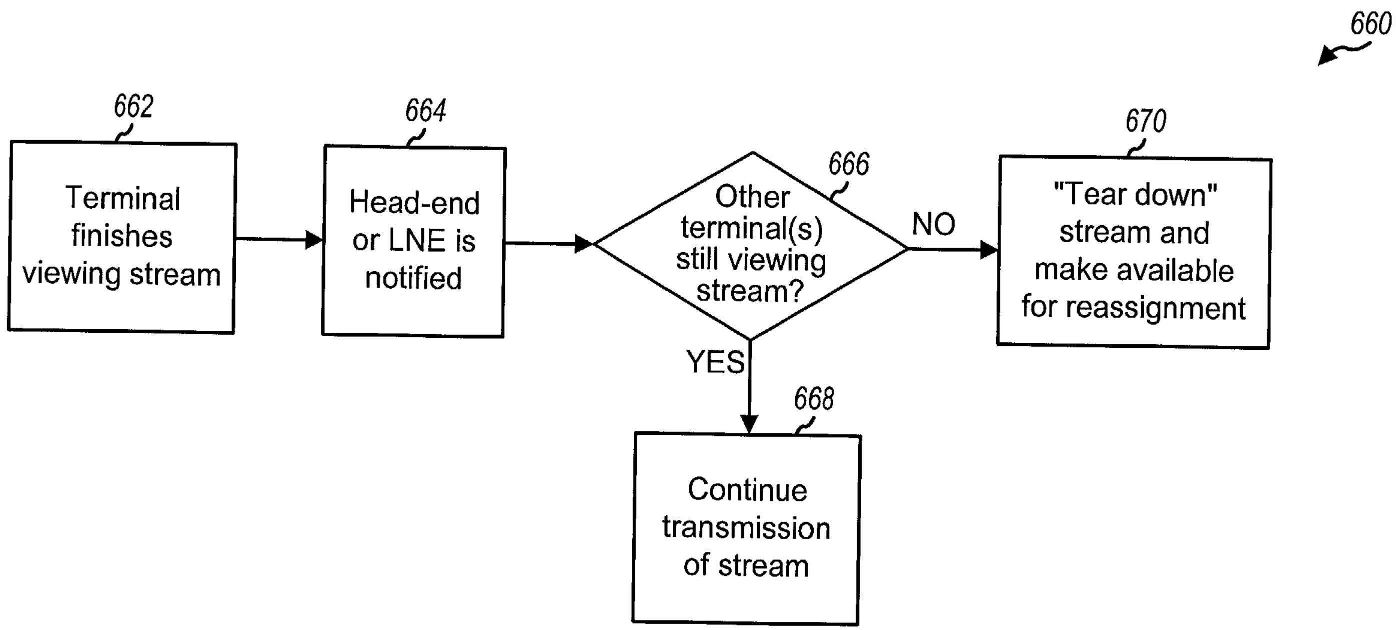

660
662
Terminal finishes viewing stream
664
Head-end or LNE is notified
666
Other terminal(s) still viewing stream?
NO
670
"Tear down" stream and make available for reassignment
YES
668
Continue transmission of stream

FIG.6C

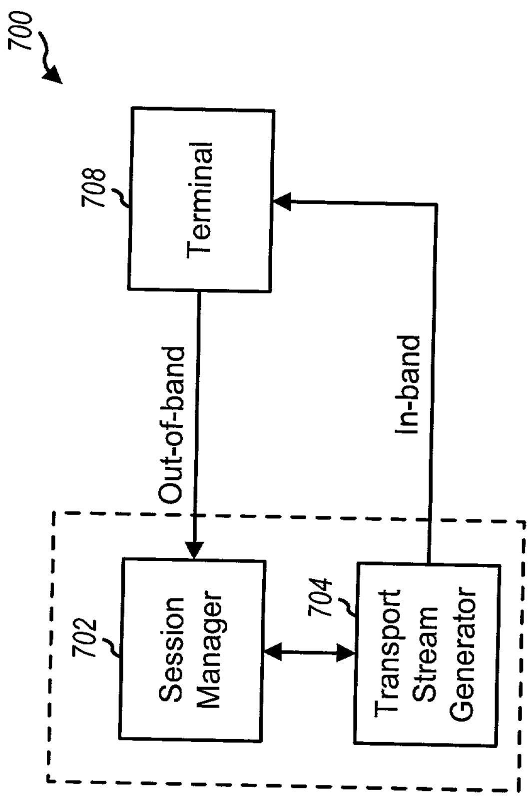
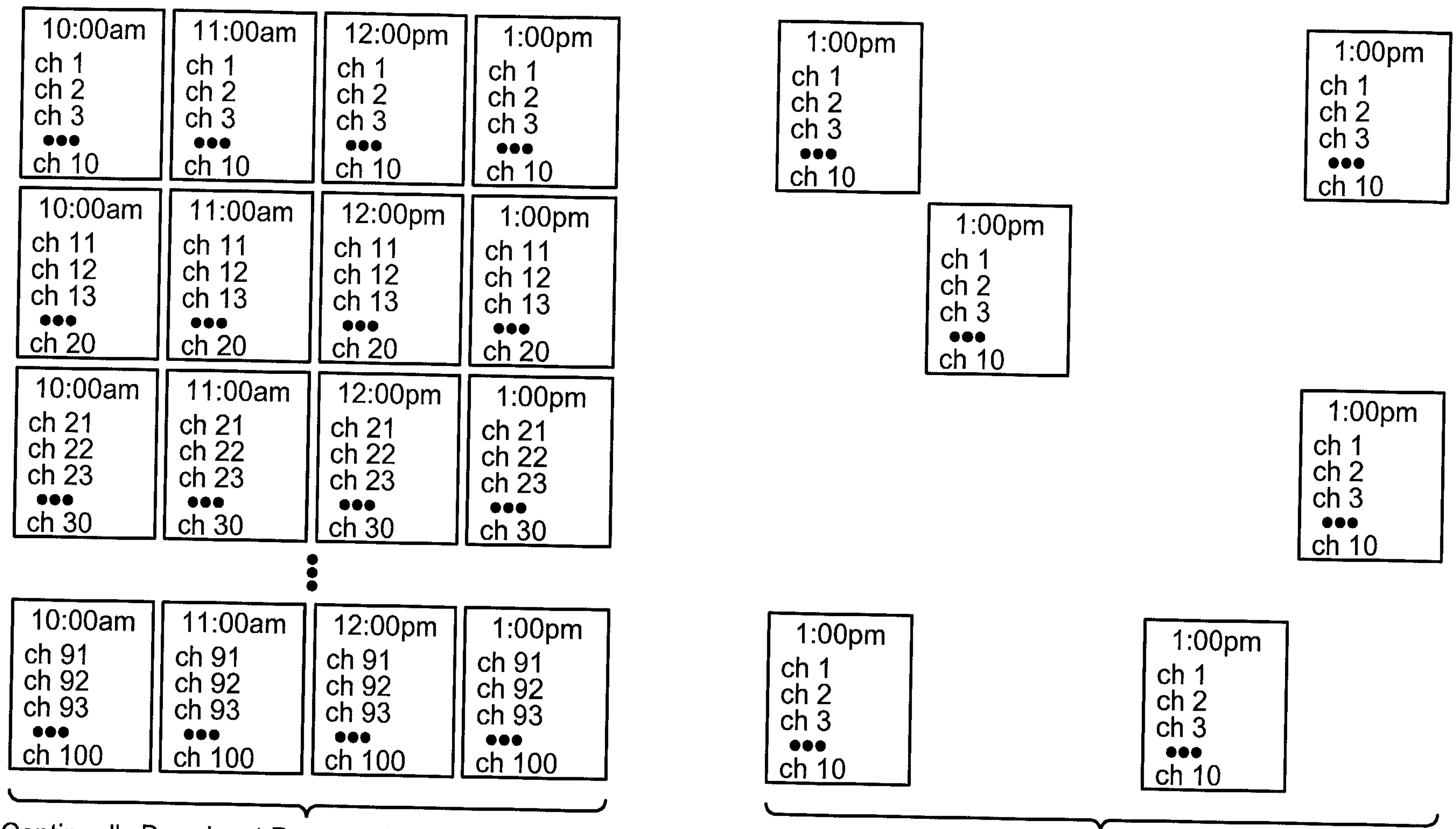

700
708
Terminal
Out-of-band
In-band
702
Session Manager
704
Transport Stream Generator

FIG.7



Continually Broadcast Pages = 40 pages.
Pages are for current time + 3 adjacent hours look-aheads in guide page matrix.
Pages are continually broadcast.

Demand-cast Pages = 30 pages.
Pages are anywhere in guide page matrix beyond continually broadcast pages.

# of pages in broadcast = # of continually broadcast pages + # of demand-cast pages

# of continually broadcast pages = (# of pages per time slot) x (# of time slots in continual broadcast)

FIG. 8

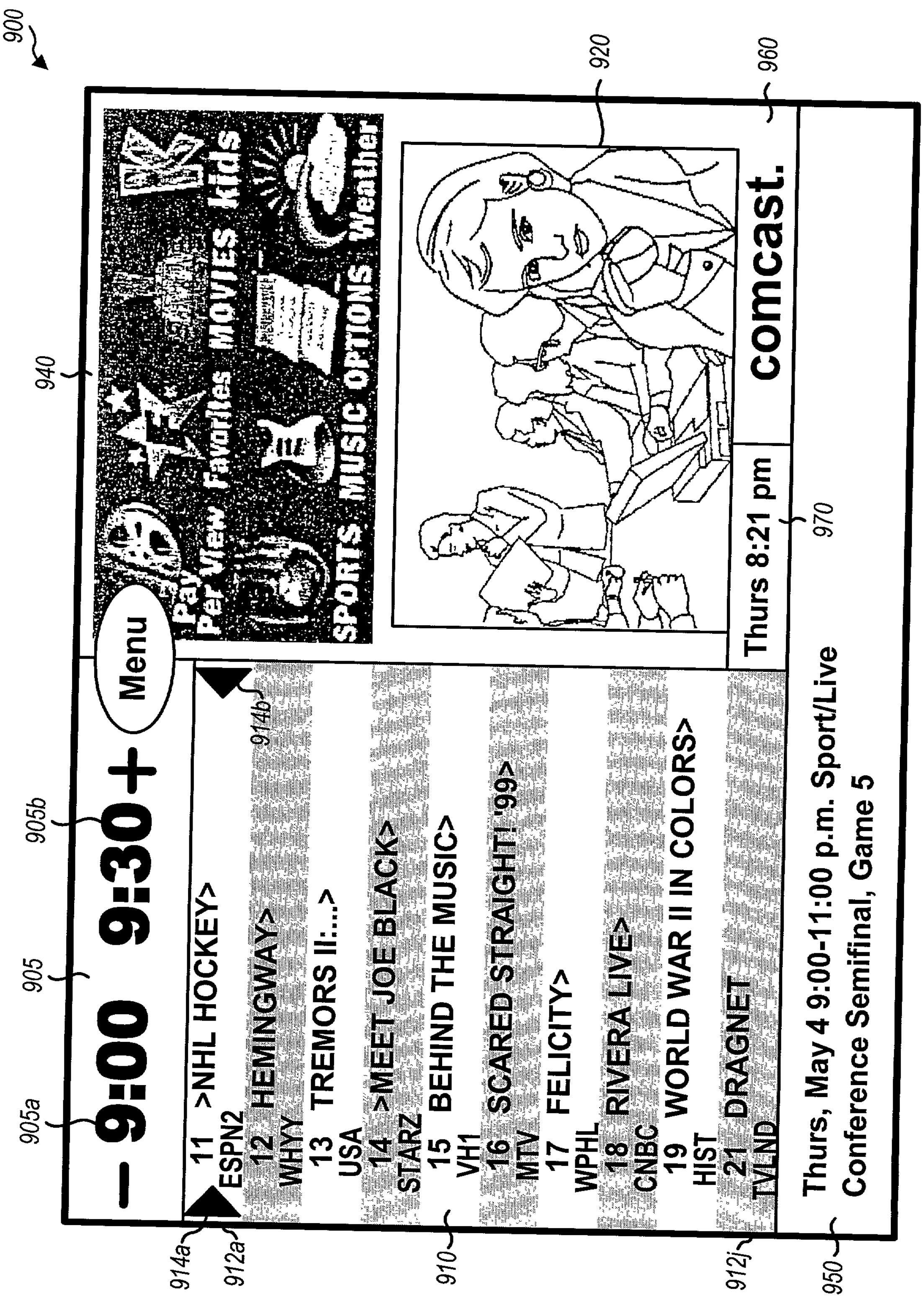

900
905a
905
905b
940
920
960
– 9:00 9:30 +
Menu
Pay Per View
Favorites
MOVIES
kids
SPORTS
MUSIC
OPTIONS
Weather
11 >NHL HOCKEY>
ESPN2
12 HEMINGWAY>
WHYY
13 TREMORS II:...>
USA
14 >MEET JOE BLACK>
STARZ
15 BEHIND THE MUSIC>
VH1
16 SCARED STRAIGHT! '99>
MTV
17 FELICITY>
WPHL
18 RIVERA LIVE>
CNBC
19 WORLD WAR II IN COLORS>
HIST
21 DRAGNET
TVLND
914a
912a
910
912j
950
914b
Thurs 8:21 pm
970
comcast.
Thurs, May 4 9:00-11:00 p.m. Sport/Live
Conference Semifinal, Game 5

FIG. 9

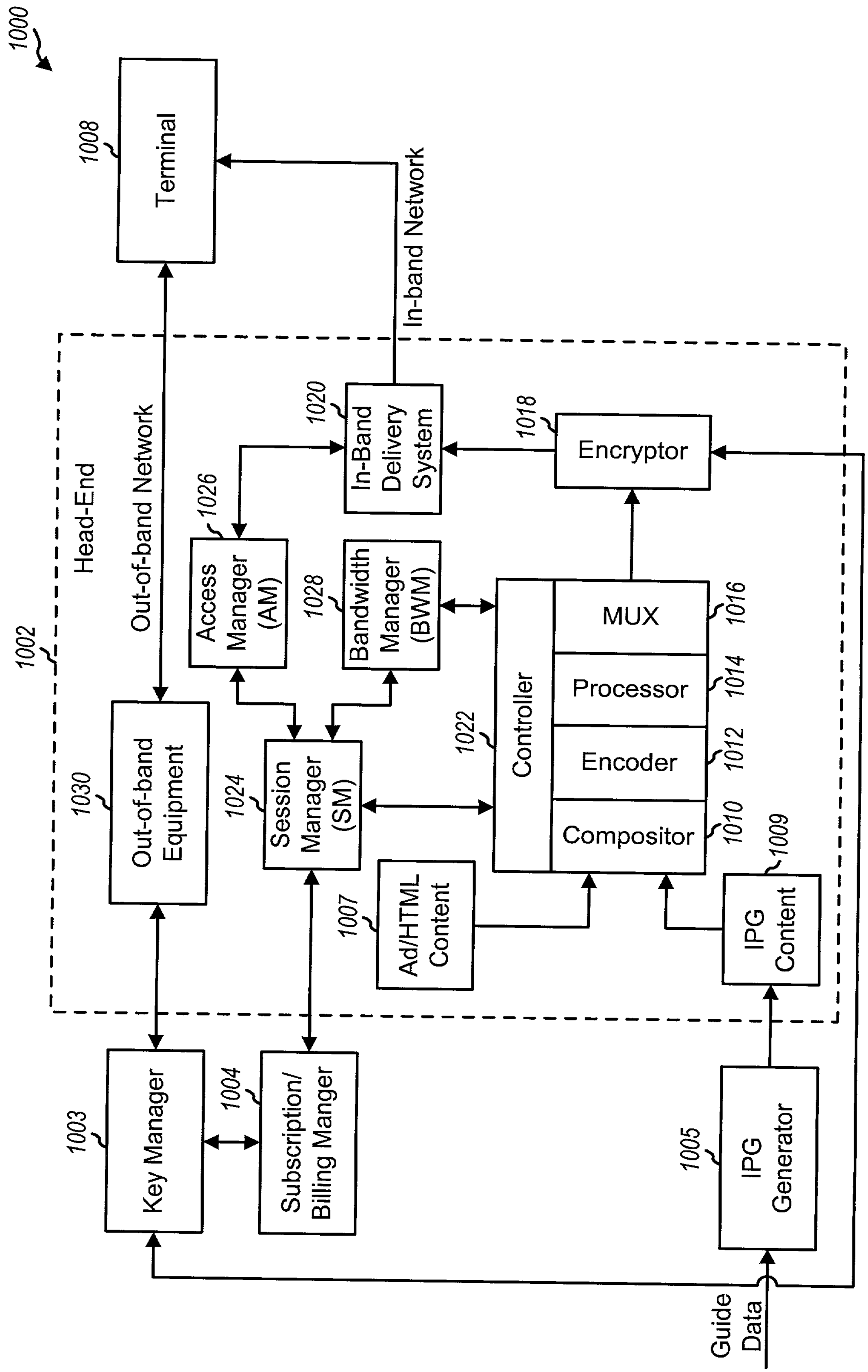

1000
1008
Terminal
In-band Network
Out-of-band Network
Head-End
1002
1020
In-Band Delivery System
1026
Access Manager (AM)
1028
Bandwidth Manager (BWM)
1018
Encryptor
1016
MUX
1014
Processor
1012
Encoder
1010
Compositor
1022
Controller
1030
Out-of-band Equipment
1024
Session Manager (SM)
1007
Ad/HTML Content
1009
IPG Content
1003
Key Manager
1004
Subscription/ Billing Manger
1005
IPG Generator
Guide Data

FIG. 10

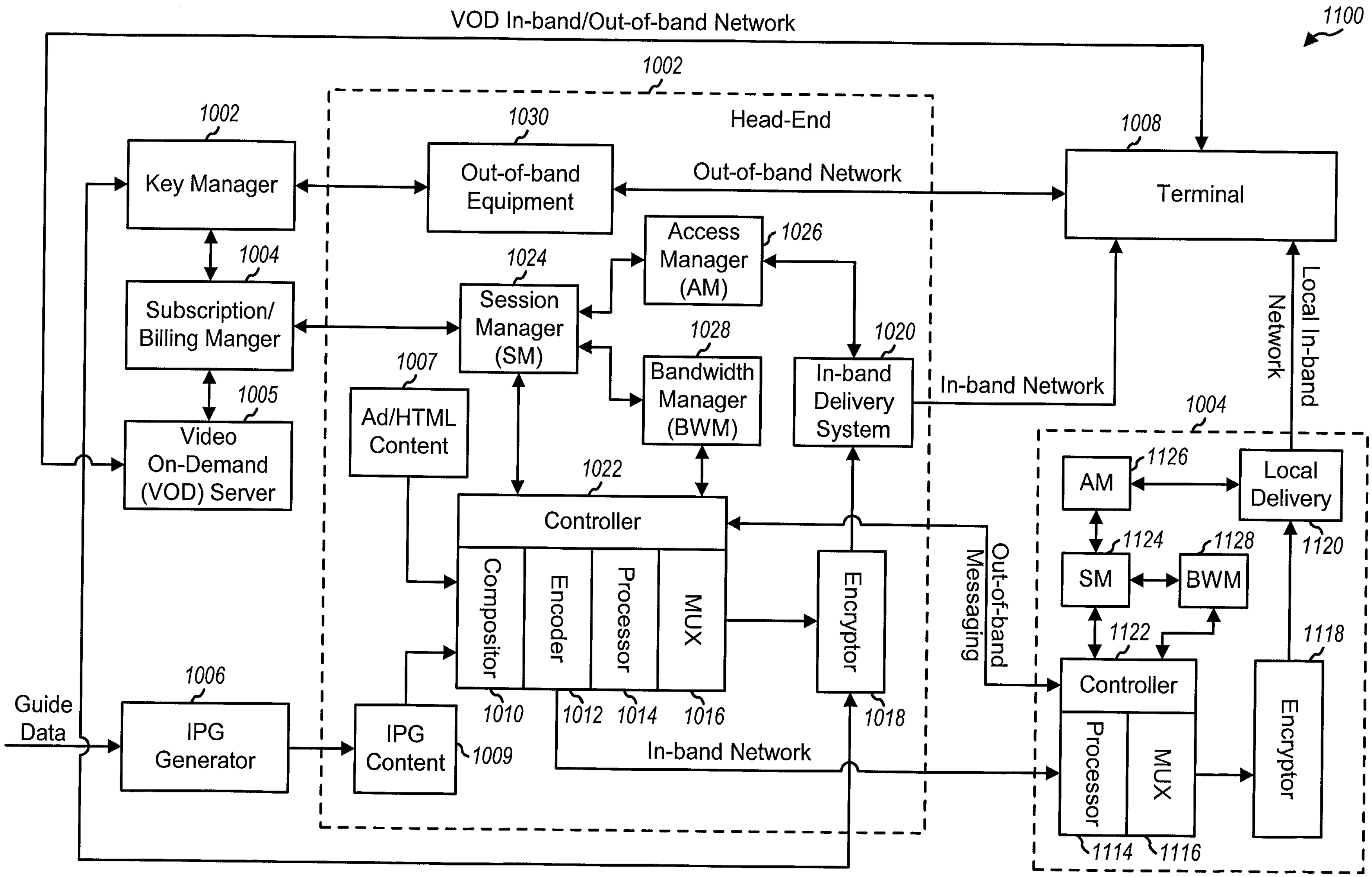
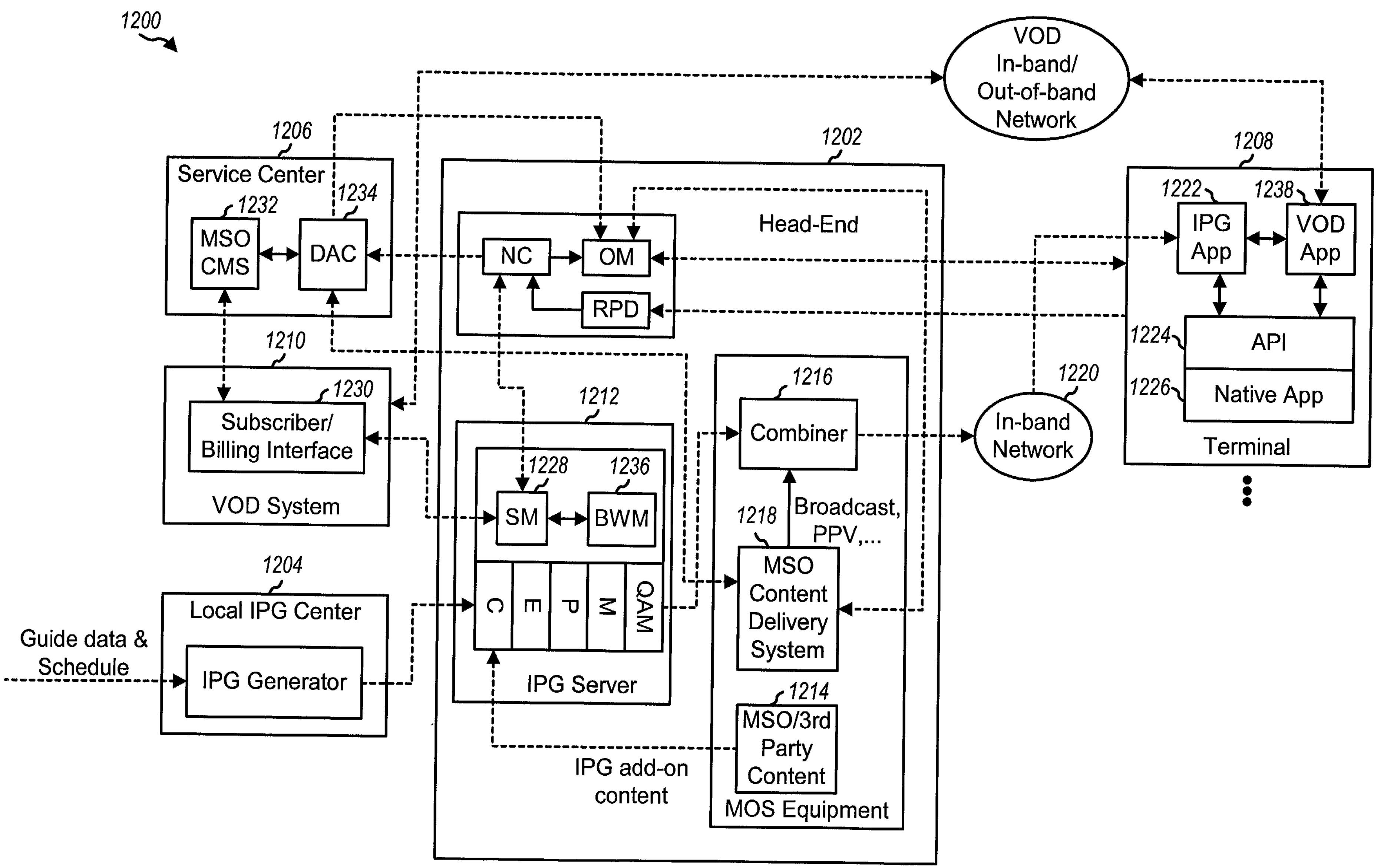

VOD In-band/Out-of-band Network
1100
1002
Head-End
1002
Key Manager
1030
Out-of-band Equipment
Out-of-band Network
1008
Terminal
1004
Subscription/ Billing Manger
1024
Session Manager (SM)
Access Manager (AM)
1026
1028
Bandwidth Manager (BWM)
1020
In-band Delivery System
In-band Network
Local In-band Network
1005
Video On-Demand (VOD) Server
1007
Ad/HTML Content
1022
Controller
Compositor
Encoder
Processor
MUX
Encryptor
1010
1012
1014
1016
1018
Out-of-band Messaging
1004
AM
1126
Local Delivery
1120
SM
1124
BWM
1128
1122
Controller
Processor
MUX
1114
1116
Encryptor
1118
Guide Data
1006
IPG Generator
IPG Content
1009
In-band Network

FIG. 11



1200
VOD In-band/ Out-of-band Network
1206
Service Center
1232
MSO CMS
1234
DAC
1202
Head-End
NC
OM
RPD
1208
1222
IPG App
1238
VOD App
1224
API
1226
Native App
Terminal
1210
1230
Subscriber/ Billing Interface
VOD System
1212
1228
SM
1236
BWM
C
E
P
M
QAM
IPG Server
1216
Combiner
1218
Broadcast, PPV,...
MSO Content Delivery System
1214
MSO/3rd Party Content
MOS Equipment
1220
In-band Network
1204
Local IPG Center
IPG Generator
Guide data & Schedule
IPG add-on content

FIG. 12

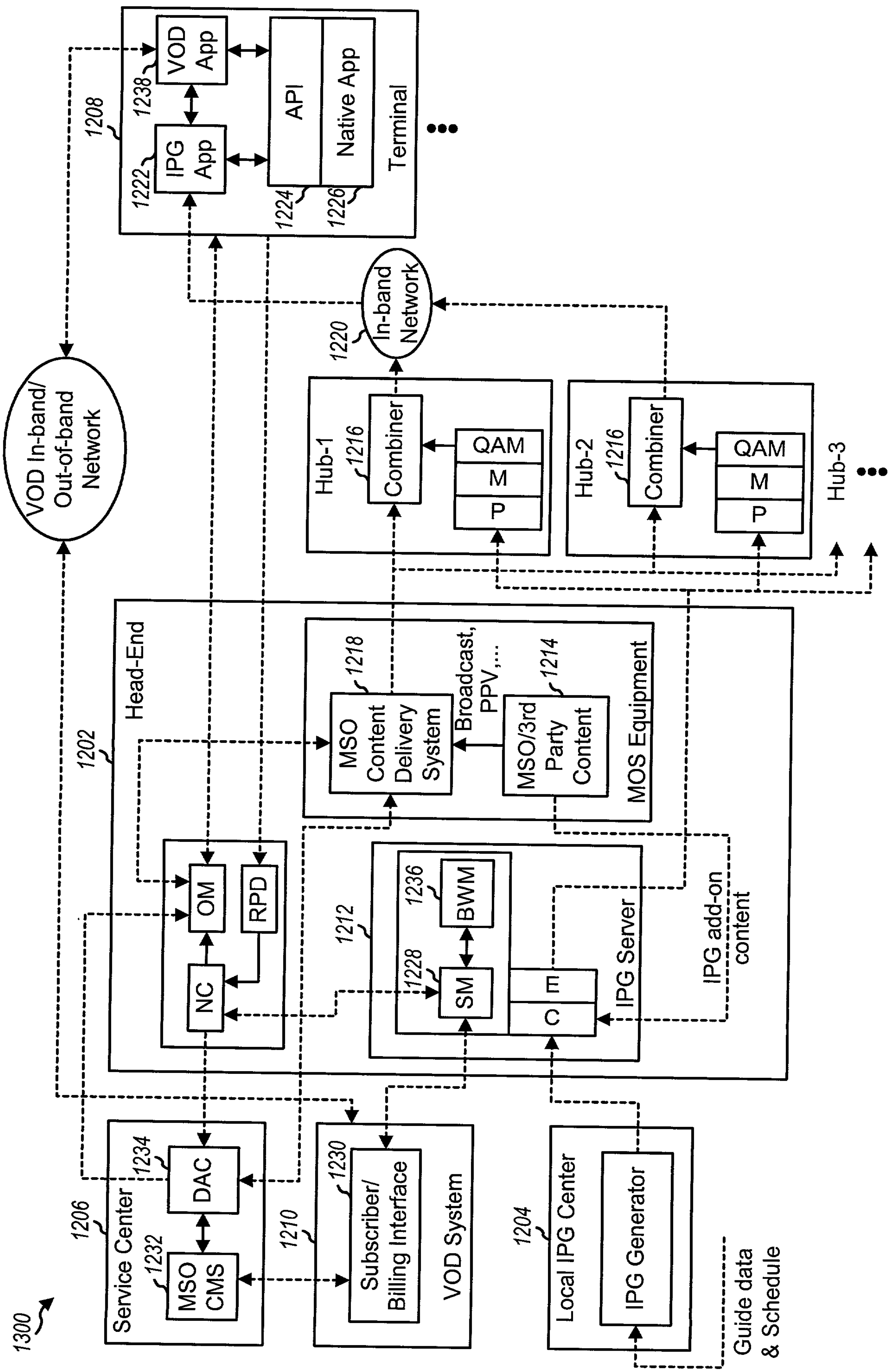

1300
VOD In-band/ Out-of-band Network
1208
1222
IPG App
1238
VOD App
API
Native App
1224
1226
Terminal
1220
In-band Network
Hub-1
1216
Combiner
QAM
M
P
Hub-2
1216
Combiner
QAM
M
P
Hub-3
1202
Head-End
1218
MSO Content Delivery System
Broadcast, PPV,...
1214
MSO/3rd Party Content
MOS Equipment
OM
RPD
NC
1212
1236
BWM
1228
SM
E
C
IPG Server
IPG add-on content
1206
Service Center
1234
DAC
1232
MSO CMS
1210
1230
Subscriber/ Billing Interface
VOD System
1204
Local IPG Center
IPG Generator
Guide data & Schedule

FIG. 13

TSG-to-Terminal Communication:

Contents of Demand-Cast Index Table

table version number (incremented when table content changes)
list of available demand-cast streams
IP address for the source TSG
MUX channel number within the source TSG
time-of-day and day-of-week

FIG. 14A

Terminal-to-SM Communication:

Message Content

demand-cast stream ID
terminal ID
IP address for the source TSG
MUX channel number within the source TSG
message information (acquisition, release, or request)

FIG. 14B

SM to-TSG Communication:

Message Content
demand-cast stream ID
MUX channel number within the source TSG
message/command (stream released, stream requested, or reset)

FIG. 14C

TSG-to-SM Communication:

Message Content
demand-cast stream ID
MUX channel number within the source TSG
IP address for the source TSG
acknowledgement (of stream release, of stream request, or of reset)

FIG. 14D

A	A	A	A	R	R	R	R
A	A	A	A	R	R	R	R
A	A	A	A	R	R	R	R
A	A	A	R	R	R	R	P
A	A	A	R	R	R	R	P

"active" demand-cast streams are in IPG multiplex

A: "acquired" streams are in MUX and in demand-cast index table

R: "released" streams are in MUX and in demand-cast index table. They can be turned into "passive" streams.

P: "passive" streams are technically released. They are not in the demand-cast index table. They are removable.

TSG Demand-cast Stream Status

FIG. 15

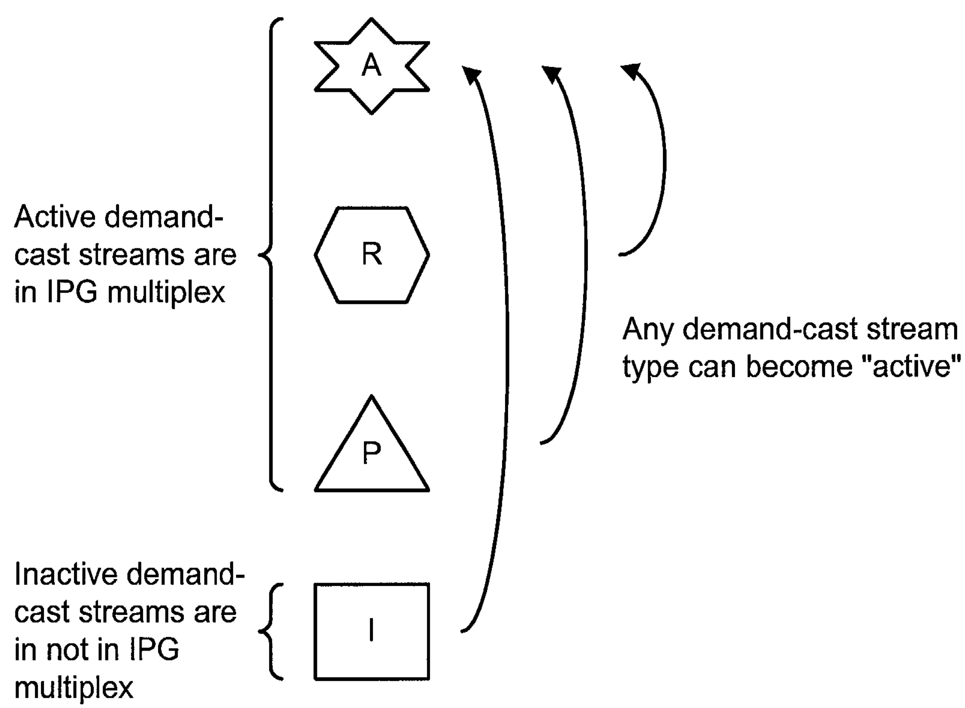

Active demand-
cast streams are
in IPG multiplex
A
R
P
I
Inactive demand-
cast streams are
in not in IPG
multiplex
Any demand-cast stream
type can become "active"

FIG.16A

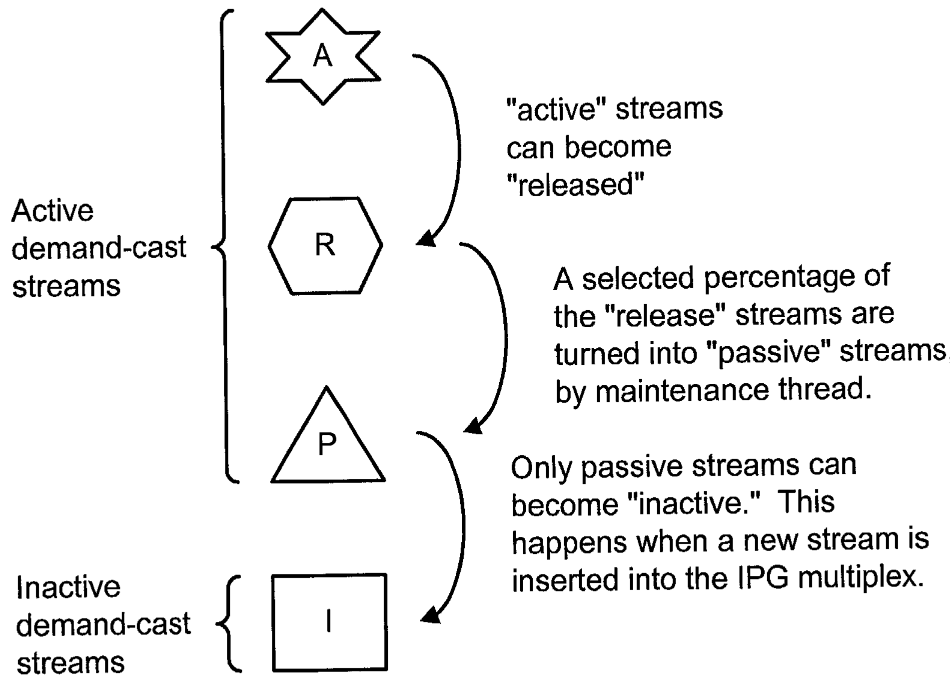

Active
demand-cast
streams
A
R
P
Inactive
demand-cast
streams
I
"active" streams
can become
"released"
A selected percentage of
the "release" streams are
turned into "passive" streams,
by maintenance thread.
Only passive streams can
become "inactive." This
happens when a new stream is
inserted into the IPG multiplex.

FIG.16B

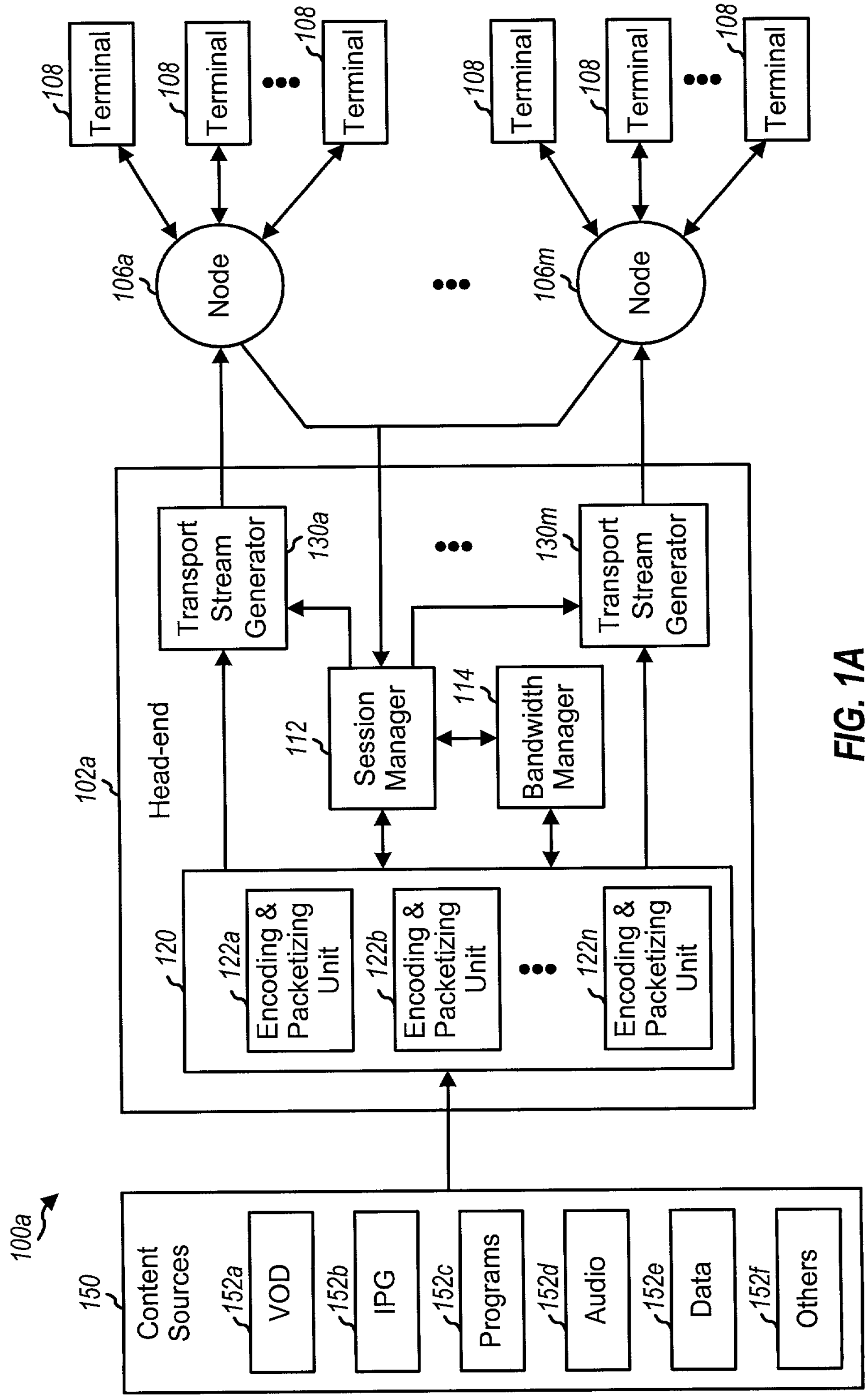

100a
150
Content Sources
152a
VOD
152b
IPG
152c
Programs
152d
Audio
152e
Data
152f
Others
102a
Head-end
120
122a
Encoding & Packetizing Unit
122b
Encoding & Packetizing Unit
122n
Encoding & Packetizing Unit
112
Session Manager
114
Bandwidth Manager
130a
Transport Stream Generator
130m
Transport Stream Generator
106a
Node
106m
Node
108
Terminal
108
Terminal
108
Terminal
108
Terminal
108
Terminal
108
Terminal

FIG. 1A

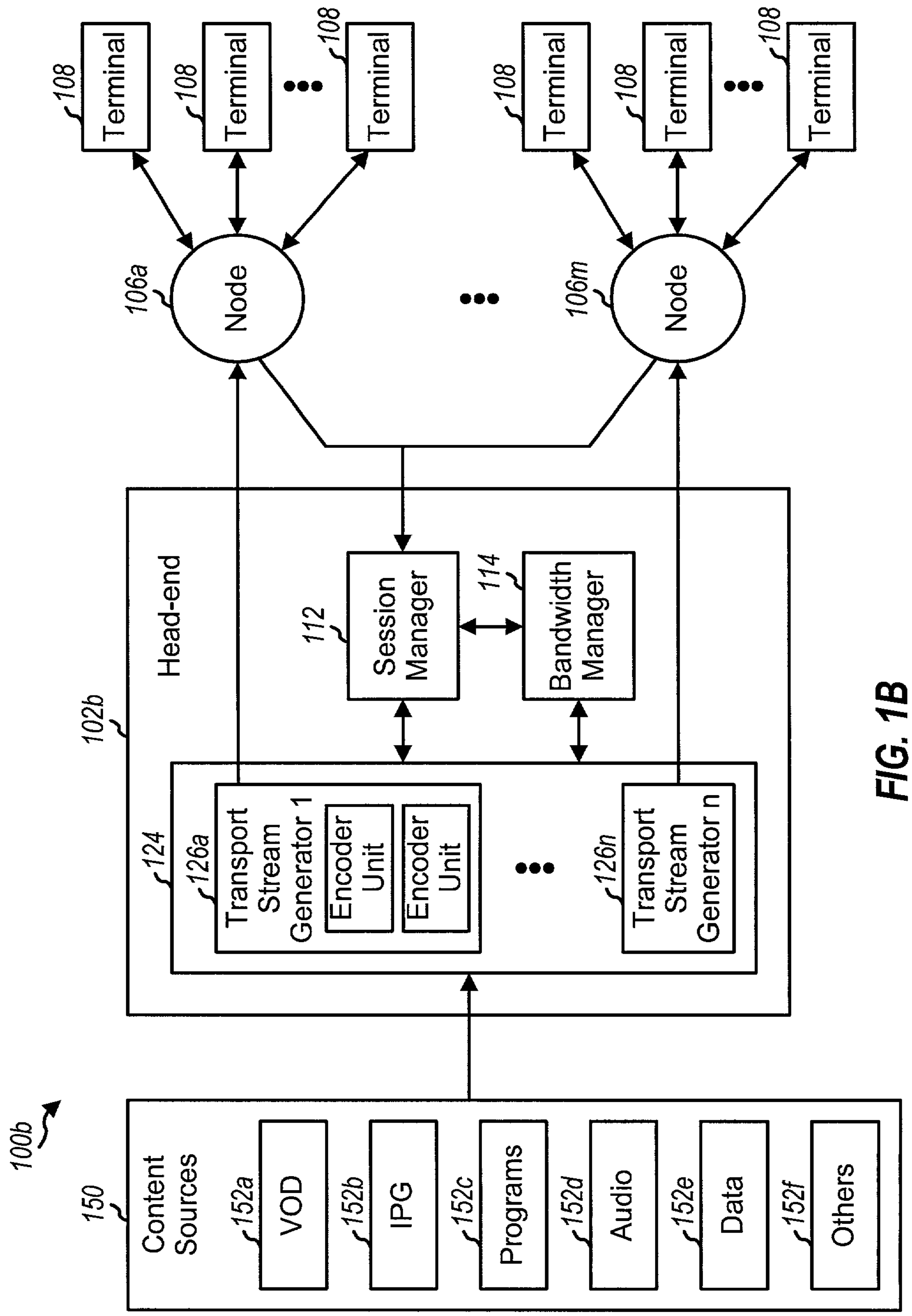

100b
150
Content Sources
152a
VOD
152b
IPG
152c
Programs
152d
Audio
152e
Data
152f
Others
102b
Head-end
124
126a
Transport Stream Generator 1
Encoder Unit
Encoder Unit
126n
Transport Stream Generator n
112
Session Manager
114
Bandwidth Manager
106a
Node
106m
Node
108
Terminal
108
Terminal
108
Terminal
108
Terminal
108
Terminal
108
Terminal

FIG. 1B

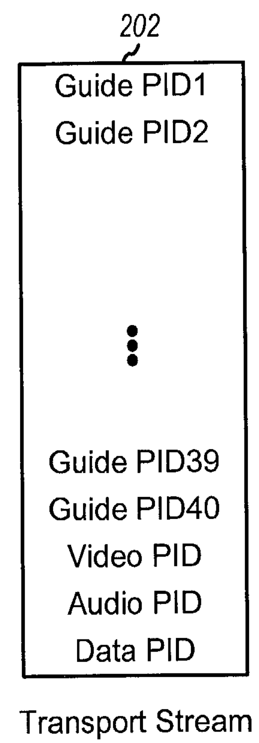

202
Guide PID1
Guide PID2
Guide PID39
Guide PID40
Video PID
Audio PID
Data PID
Transport Stream

FIG. 2A

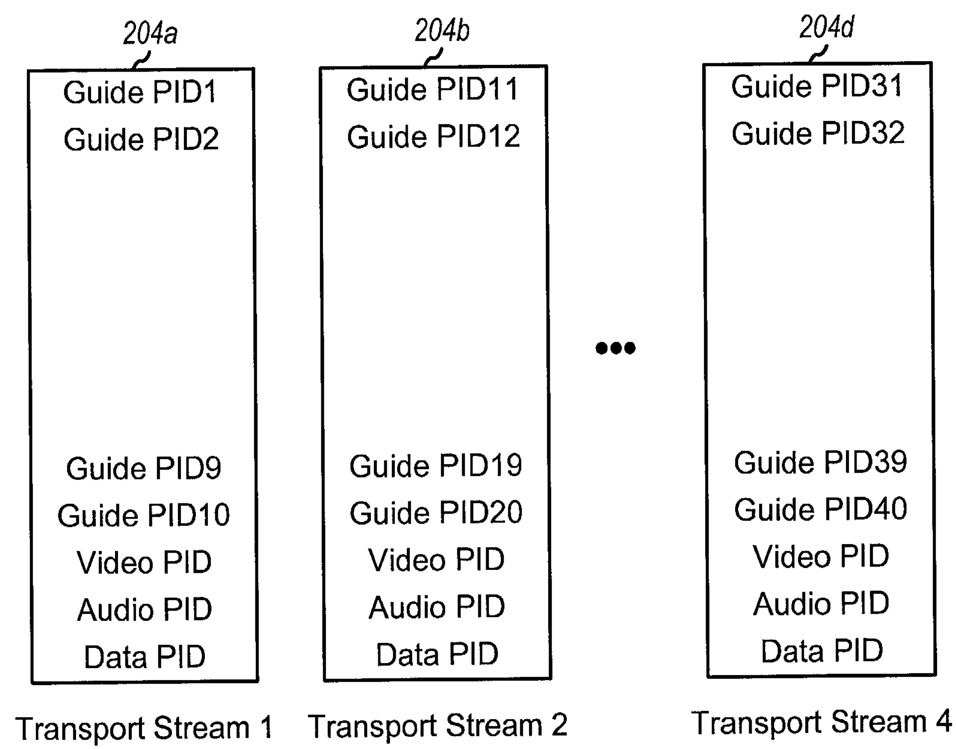

204a
Guide PID1
Guide PID2
Guide PID9
Guide PID10
Video PID
Audio PID
Data PID
Transport Stream 1
204b
Guide PID11
Guide PID12
Guide PID19
Guide PID20
Video PID
Audio PID
Data PID
Transport Stream 2
204d
Guide PID31
Guide PID32
Guide PID39
Guide PID40
Video PID
Audio PID
Data PID
Transport Stream 4

FIG. 2B

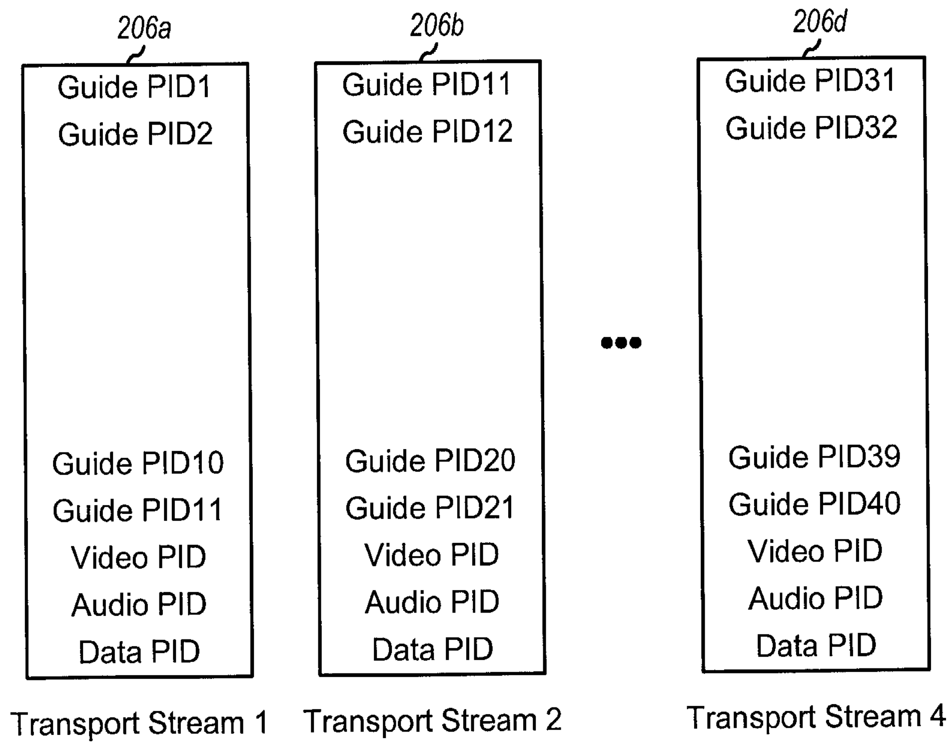

206a
Guide PID1
Guide PID2
Guide PID10
Guide PID11
Video PID
Audio PID
Data PID
Transport Stream 1
206b
Guide PID11
Guide PID12
Guide PID20
Guide PID21
Video PID
Audio PID
Data PID
Transport Stream 2
206d
Guide PID31
Guide PID32
Guide PID39
Guide PID40
Video PID
Audio PID
Data PID
Transport Stream 4

FIG. 2C

DECLARATION

As a below named inventor, I declare that:

My residence, post office address and citizenship are as stated below next to my name; I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: **METHOD AND SYSTEM FOR MULTICAST USING MULTIPLE TRANSPORT STREAMS** the specification of which was filed on **10/4/00** as Application Serial No. **Unassigned.**

I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I acknowledge the duty to disclose information that is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I claim no foreign priority benefits under Title 35, United States Code, Section 119.

I claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below.

Application No.	Date of Filing

I claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, Section 1.56 which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

Application No.	Date of Filing	Status
09/293,535	April 15, 1999	Pending
09/384,394	August 27, 1999	Pending
09/428,066	October 27, 1999	Pending
09/466,990	December 10, 1999	Pending
09/524,854	March 14, 2000	Pending
09/539,228	March 30, 2000	Pending

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

<table>
<tr><td>Full Name of Inventor 2</td><td>Last Name:
GORDON</td><td>First Name:
DONALD</td><td colspan="2">Middle Name or Initial:
F.</td></tr>
<tr><td>Residence & Citizenship:</td><td>City:
Los Altos</td><td>State/Foreign Country:
CA</td><td colspan="2">Country of Citizenship:
US</td></tr>
<tr><td>Post Office Address:</td><td>Post Office Address:
170 Formway Court</td><td>City:
Los Altos</td><td>State/Country:
CA</td><td>Postal Code:
94022</td></tr>
<tr><td>Signature & Date:</td><td colspan="2">Signature of Inventor :</td><td colspan="2">Date:</td></tr>
</table>

Full Name of Inventor 1	Last Name: **FEINBERG**	First Name: **BRIAN**	Middle Name or Initial:	
Residence & Citizenship:	City: **Cupertino**	State/Foreign Country: **CA**	Country of Citizenship: **US**	
Post Office Address:	Post Office Address: **19782 Bixby Drive**	City: **Cupertino**	State/Country: **CA**	Postal Code: **95014**
Signature & Date:	Signature of Inventor :		Date:	

Full Name of Inventor 3	Last Name: **GERSHTEIN**	First Name: **EUGENE**	Middle Name or Initial:	
Residence & Citizenship:	City: **Redwood City**	State/Foreign Country: **CA**	Country of Citizenship: **US**	
Post Office Address:	Post Office Address: **1418 Valota Road**	City: **Redwood City**	State/Country: **CA**	Postal Code: **94061**
Signature & Date:	Signature of Inventor :		Date:	

Full Name of Inventor 4	Last Name: **BAYRAKERI**	First Name: **SADIK**	Middle Name or Initial:	
Residence & Citizenship:	City: **Foster City**	State/Foreign Country: **CA**	Country of Citizenship: **TR**	
Post Office Address:	Post Office Address: **733 Shell Boulevard #104**	City: **Foster City**	State/Country: **CA**	Postal Code: **94404**
Signature & Date:	Signature of Inventor :		Date:	

Full Name of Inventor 5	Last Name: **CAMITO**	First Name: **JOHN**	Middle Name or Initial: **P.**	
Residence & Citizenship:	City: **Redwood City**	State/Foreign Country: **CA**	Country of Citizenship: **US**	
Post Office Address:	Post Office Address: **907 Pleasant Hill Road**	City: **Redwood City**	State/Country: **CA**	Postal Code: **94061**
Signature & Date:	Signature of Inventor :		Date:	

Full Name of Inventor 6	Last Name: **LUDVIG**	First Name: **EDWARD**	Middle Name or Initial: **A.**	
Residence & Citizenship:	City: **Redwood City**	State/Foreign Country: **CA**	Country of Citizenship: **CA**	
Post Office Address:	Post Office Address: **931 Canyon Road**	City: **Redwood City**	State/Country: **CA**	Postal Code: **94061**
Signature & Date:	Signature of Inventor :		Date:	